



Video – Parte 4a

Stabilizzazione



Stabilizzazione

- Un sistema di stabilizzazione dell'immagine ha come scopo quello di **rimuovere i movimenti** da una sequenza di immagini
 - **Padding**: movimenti intenzionali
 - **Jitter**: movimenti non intenzionali
- Sistemi di stabilizzazione analogici
 - Accelerometri, giroscopi, sensori di velocità angolare, ammortizzatori meccanici, ...
- Analizzeremo **sistemi di stabilizzazione digitali (DIS)**



Sistemi di Stabilizzazione Digitale

- Esempio:
 - La linea rettilinea, a differenza di quella curva, indica che la sequenza di frame è stabile
 - Crop!





Sistemi di Stabilizzazione Digitale

- Distinguiamo i Digital Image Stabilization systems in:
 - DIS real-time
 - DIS post-processing

- 3 fasi principali:
 1. Stima del movimento
 2. Filtraggio (*correzione*) del movimento
 3. Deformazione (*post-processing*) dell'immagine



Sistemi di Stabilizzazione Digitale

Fase 1 – Stima del movimento

- Vedi Video – Parte 3 =)
 - Algoritmi di Block-Matching (BMA)
 - Criterio DFD
 - Algoritmi di Features Extraction



Sistemi di Stabilizzazione Digitale

Fase 2 – Filtraggio del movimento

- In questa fase si vuole riuscire a distinguere i movimenti intenzionali (*padding*) da quelli casuali (*jitter*), per poi tentare di correggere i secondi
- Vedremo 3 algoritmi di motion filtering:
 - **MVI**: *Motion Vector Integration*
 - **FPS**: *Frame Position Smoothing*
 - **Filtro Kalman**



Fase 2 – Filtraggio del movimento

Motion Vector Integration (1)

- Idea di base:
 - Usare i metodi della Fase 1 per stimare dei MV locali (**LMV**) affidabili
 - Calcolare il MV globale (**GMV**)
 - Il GMV può essere calcolato a partire dai LMV
 - Chiamiamo $V_A(n)$ il GMV dell'anchor frame
 - “Integrare” (stimare) un MV che rappresenterà la variazione di movimento rispetto al target frame
 - $V_I(n) = \delta \cdot V_I(n - 1) + V_A(n)$
 - $0 < \delta \leq 1$ “*damping factor*”: pesa il grado di casualità del movimento presente (di solito oscilla fra 0.875 e 0.995)



Fase 2 – Filtraggio del movimento

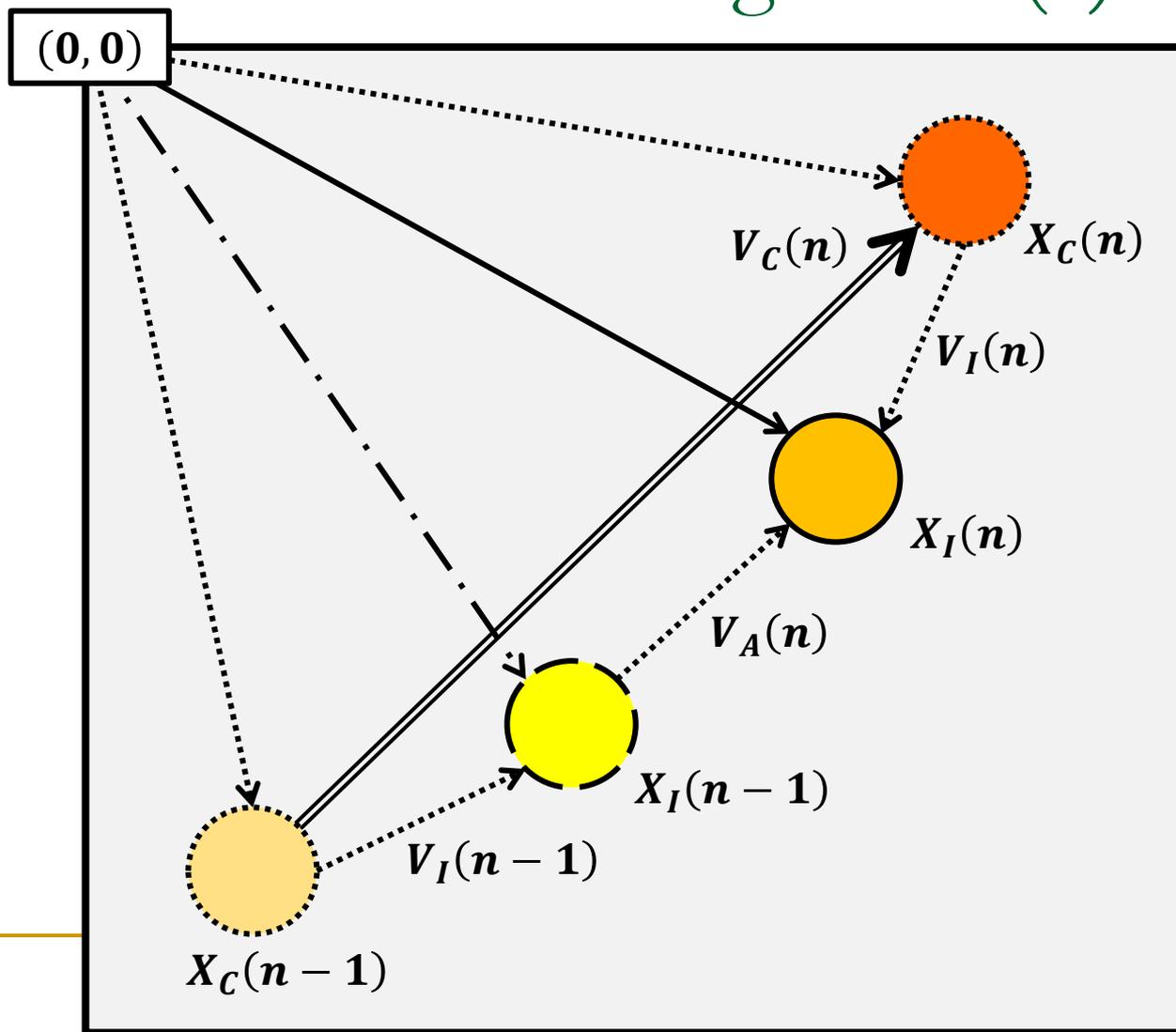
Motion Vector Integration (2)

- Ridefiniamo $V_A(n)$ rispetto al punto X :
 - $V_A(n) = X_I(n) - X_I(n - 1)$
- Rispetto alla posizione Iniziale, X dovrebbe trovarsi nella posizione Corretta:
 - $X_C(n) = X_I(n) - V_I(n)$ *nell'anchor frame*
 - $X_C(n - 1) = X_I(n - 1) - V_I(n - 1)$ *nel target frame*
- Infine, definiamo il vettore di Correzione:
 - $V_C(n) = X_C(n) - X_C(n - 1) =$
 $= V_A(n) - V_I(n) + V_I(n - 1)$



Fase 2 – Filtraggio del movimento Motion Vector Integration (3)

Un esempio
di MVI





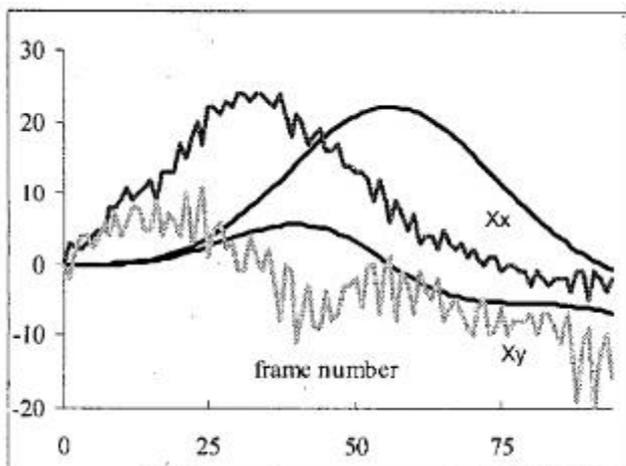
Fase 2 – Filtraggio del movimento Frame Position Smoothing (1)

- Idea di base per calcolare il MV correttore:
 - $V_C(n) = X_{LPF}(n) - X_A(n)$
 - $X_A(n)$: posizione (assoluta, rispetto al primo frame)
 - $X_{LPF}(n)$: posizione corretta, ottenuta:
 - passando al dominio delle frequenze (DFT)
 - confrontando la fase rispetto ai frame precedenti
 - applicando un filtro passa basso (LPF), ad esempio una gaussiana

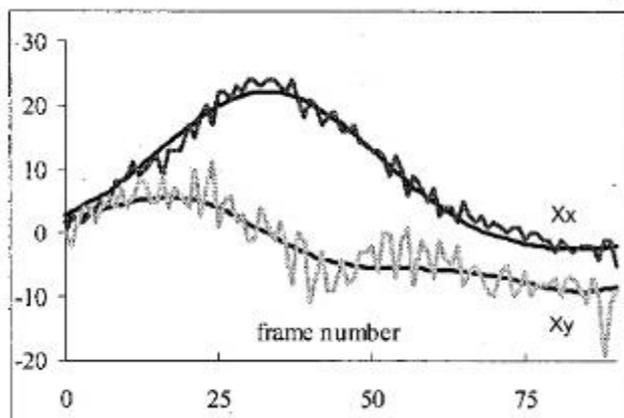


Fase 2 – Filtraggio del movimento Frame Position Smoothing (2)

- Il filtraggio nel dominio delle frequenze ha il problema di richiedere un'elaborazione offline
- In un eventuale dispositivo real-time si deve introdurre un delay, ad esempio:



$$V_C(n) = X_{LPF}(n + 25) - X_A(n)$$



*Scostamenti assoluti
(rispetto al primo frame)
nelle direzioni X e Y.
Senza (a sinistra) e con
(a destra) ritardo.*



Fase 2 – Filtraggio del movimento

Filtro di Kalman (1)

- Strumento molto potente che:
 - Prende in input una serie di misure osservate nel tempo
 - Tiene conto di un eventuale rumore casuale
 - **Predizione:** fissato un tempo t stima la misura Z_{t+1}
 - **Correzione:** osservata una misura può correggerla, eventualmente basandosi su una predizione



Fase 2 – Filtraggio del movimento

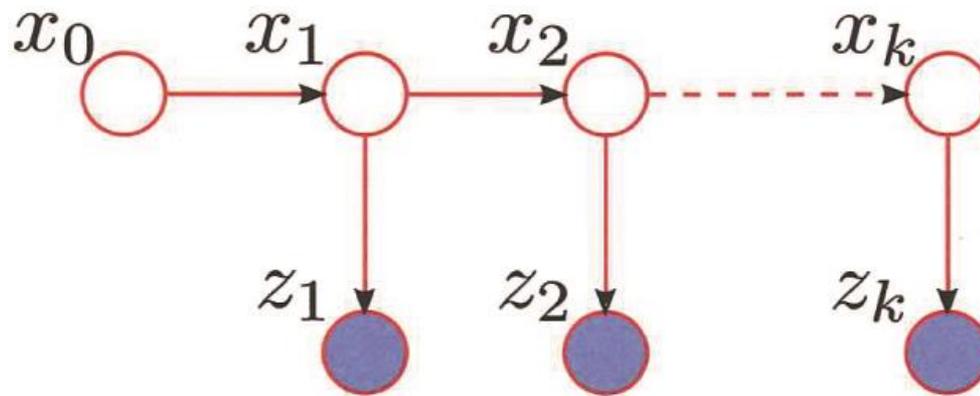
Filtro di Kalman (2)

- La versione discreta del filtro si basa su due *equazioni differenziali stocastiche (lineari)*:
 - Nel processo stocastico, lo stato x_k è dato da:
 - $x_t = Ax_{t-1} + Bu_t + w_t$
 - La misura z_k di x_k è data da:
 - $z_t = Hx_t + v_t$
 - Dove:
 - t : istante temporale; A, B, H : modelli transazionali; w, v : rumore gaussiano; u : controllo utente



Fase 2 – Filtraggio del movimento

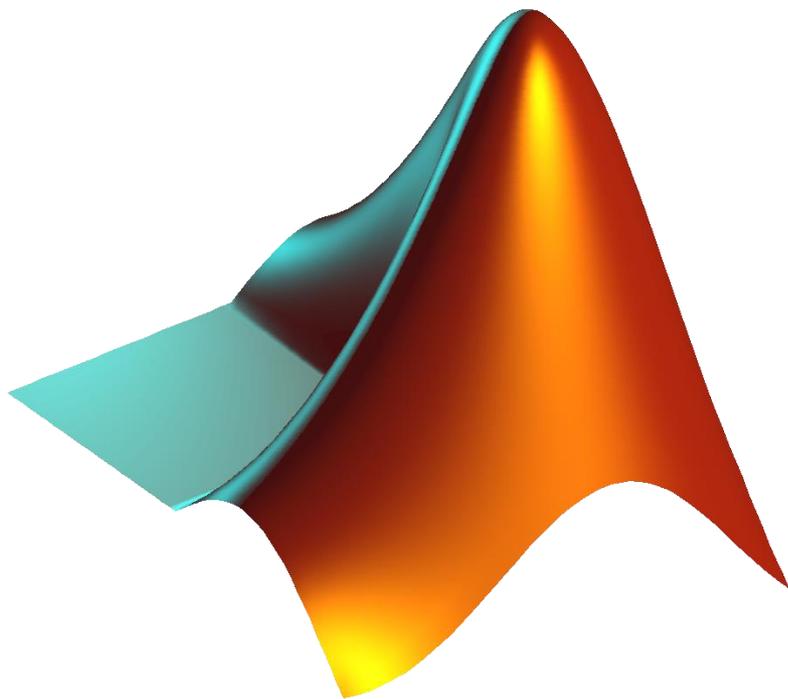
Filtro di Kalman (3)



- Affiancando il filtro di Kalman ad un tracciatore Bayesiano possiamo stimare le probabilità:
 - **Predizione:** $P(x_t | z_{t-1})$
 - **Correzione (update):** $P(x_t | z_t)$



Stabilizzazione Filtraggio del Movimento



- **MotionDetectionBGScript.m**
- **KalmanPredictScript.m**
 - Rilevare il movimento tramite sottrazione dello sfondo
 - Una volta individuato l'oggetto in movimento lo tratteremo con un filtro di Kalman
- **StabilizationFeatureScript.m**
- **VideoStabilizationScript.m**



Fase 3 – Post-processing: Deformazione dell'immagine (1)

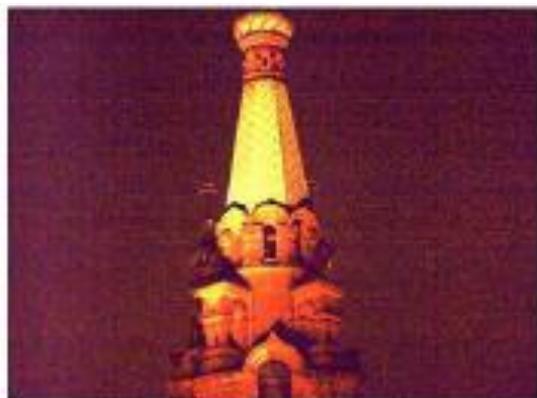
- Una fase di post-processing dei frame è necessaria perché il crop dei frame fa perdere informazioni
 - Lo scopo principale è migliorare la qualità dei frame





Fase 3 – Post-processing: Deformazione dell'immagine (2)

- Si può agire anche sui valori di intensità nei pixel, qualora l'esposizione del video non sia uniforme



Sottoesposizione



Sovraesposizione



Stabilizzata