# Analisi di Immagini e Video (Computer Vision)

Giuseppe Manco

# Outline

- Activity Recognition
  - Task
  - Datasets
- Approcci
  - 3D-CNN
  - RNN-CNN
  - Optical Flow

# Crediti

- Slides adattate da altri corsi:
  - Ettore Ritacco (CS Unical)
  - Joseph Redmon (CS Washington EDU)
  - Rob Fergus (CS NYU EDU

# Activity Recognition

- Gli algoritmi che abbiamo visto finora si applicano al dominio spaziale
  - Classificazione
  - Segmentazione
  - Scene understanding
  - …
- Che significa includere il dominio temporale?
  - Immagini che fluiscono lungo l'asse temporale
  - video
- Desiderata
  - Catturare le caratteristiche del «movimento» (motion) e sfruttarle per la classificazione
  - In maniera computazionalmente gestibile

# Esempio

- Sapreste riconoscere da un singolo frame lo stile di nuotata?

# Perché è difficile

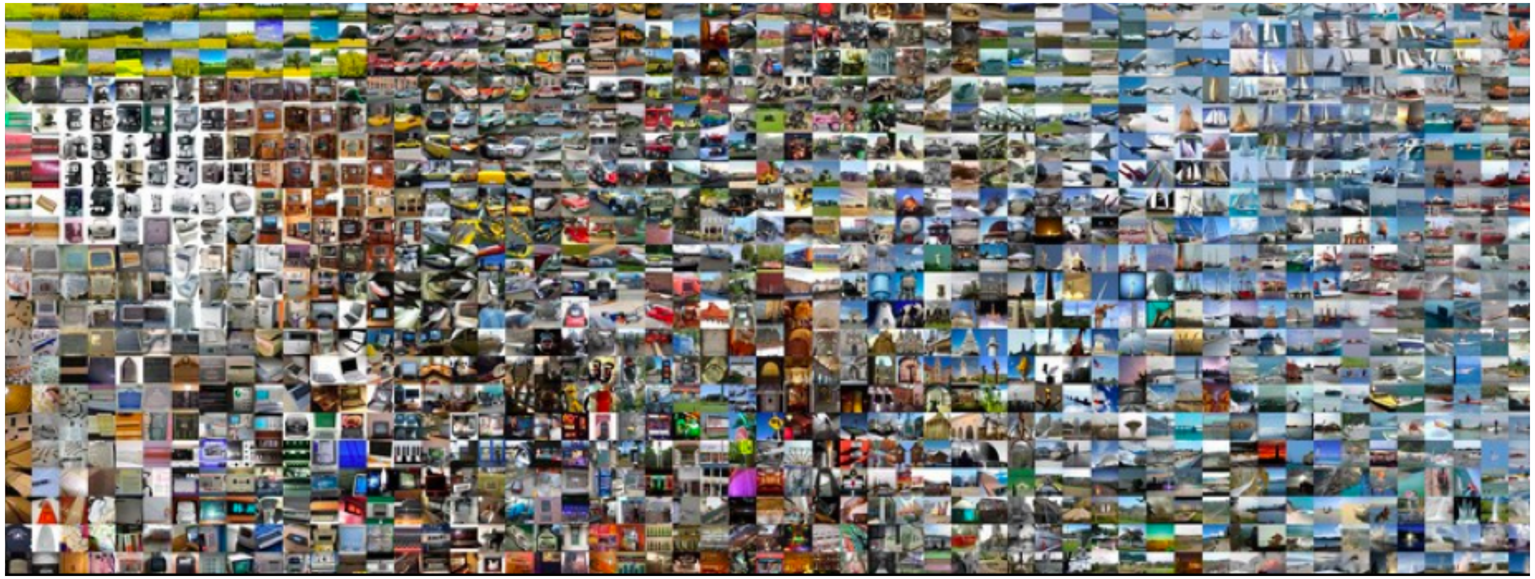
- Alto costo computazionale
  - In un video la quantità di immagini è alta (~25fps)
- Necessità di catturare i contesti short-term (pochi frame) e long-term (secondi, minuti, …)
- Difficoltà di reperire dati di training

# Datasets per activity recognition

- UCF-101
  - 10k videos
- HMDB-51
  - 5k videos

# Datasets per activity recognition

- Kinetics-400
  - 300k videos
  - 10s clips
- Human action classification
  - 400 human action classes



# Confronto

| Dataset | Year | Actions | Clips | Total | Videos |
|---|---|---|---|---|---|
| HMDB-51 [15] | 2011 | 51 | min 102 | 6,766 | 3,312 |
| UCF-101 [20] | 2012 | 101 | min 101 | 13,320 | 2,500 |
| ActivityNet-200 [3] | 2015 | 200 | avg 141 | 28,108 | 19,994 |
| Kinetics | 2017 | 400 | min 400 | 306,245 | 306,245 |
| **Kinetics-600** | **2018** | **600** | **min 450** | **500,000** | **500,000** |
| **Kinetics-700** | **2019** | **700** | **min 450** | **650,000** | **650,000** |

# Youtube8M

- large-scale labeled video dataset
  - high-quality machine-generated annotations from a diverse vocabulary of 3,800+ visual entities
  - scale and diversity

**6.1 Million**
Video IDs

**350,000**
Hours of Video

**2.6 Billion**
Audio/Visual Features

**3862**
Classes

**3.0**
Avg. Labels / Video

The videos are sampled uniformly to preserve the diverse distribution of popular content on YouTube, subject to a few constraints selected to ensure dataset quality and stability:

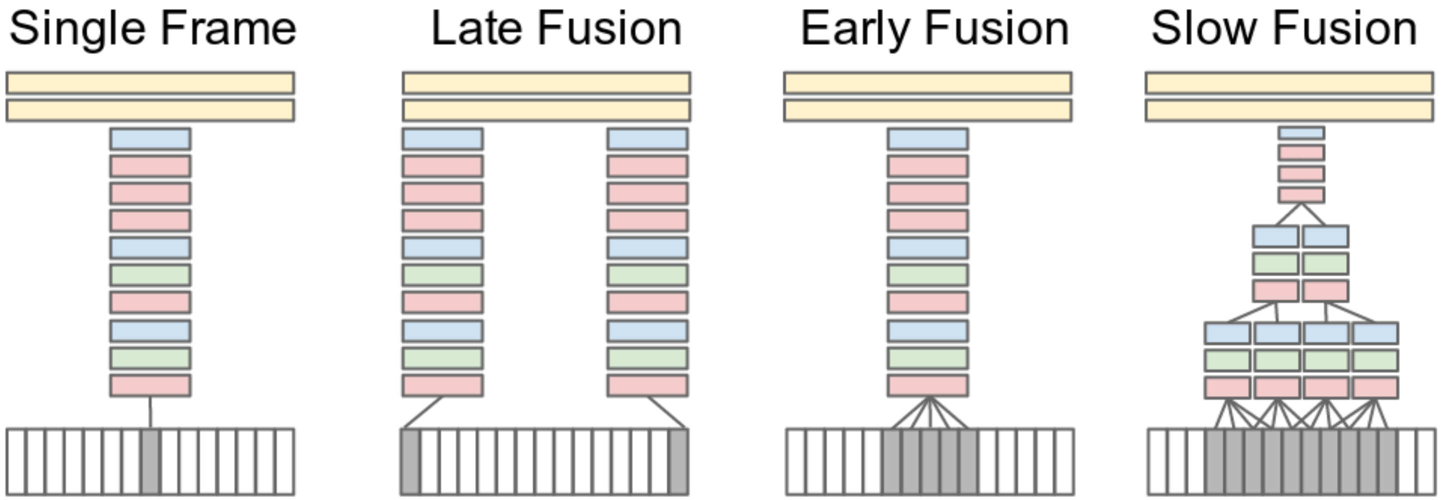
- Each video must be public and have at least 1000 views
- Each video must be between 120 and 500 seconds long
- Each video must be associated with at least one entity from our target vocabulary
- Adult & sensitive content is removed (as determined by automated classifiers)

# Altri datasets

- Sports-1M
  - 400 sport classes
- Something-something
  - 174 classes
- HACS
  - 200 classes, positive/negative samples

# 2D Convolution non funziona!

- Varie possibilità di combinare i frame
- Accuratezza scarsa
  - Non cattura feature spazio-temporali



# 2D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -8 | 1 |
| 1 | 1 | 1 |

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 0 |
| 0 | 2 | -7 | 2 | 0 |
| 0 | 3 | -6 | 3 | 0 |
| 0 | 2 | -7 | 2 | 0 |
| 0 | 1 | 1 | 1 | 0 |

# 2D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -8 | 1 |
| 1 | 1 | 1 |

| | | | | |
|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 |
| 1 | -7 | 2 | 1 | 0 |
| 1 | 2 | -6 | 2 | 1 |
| 0 | 1 | 2 | -7 | 1 |
| 0 | 0 | 1 | 1 | 1 |

# 2D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -8 | 1 |
| 1 | 1 | 1 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 2 | 3 | 2 | 1 |
| 1 | -7 | -6 | -7 | 1 |
| 1 | 2 | 3 | 2 | 1 |
| 0 | 0 | 0 | 0 | 0 |

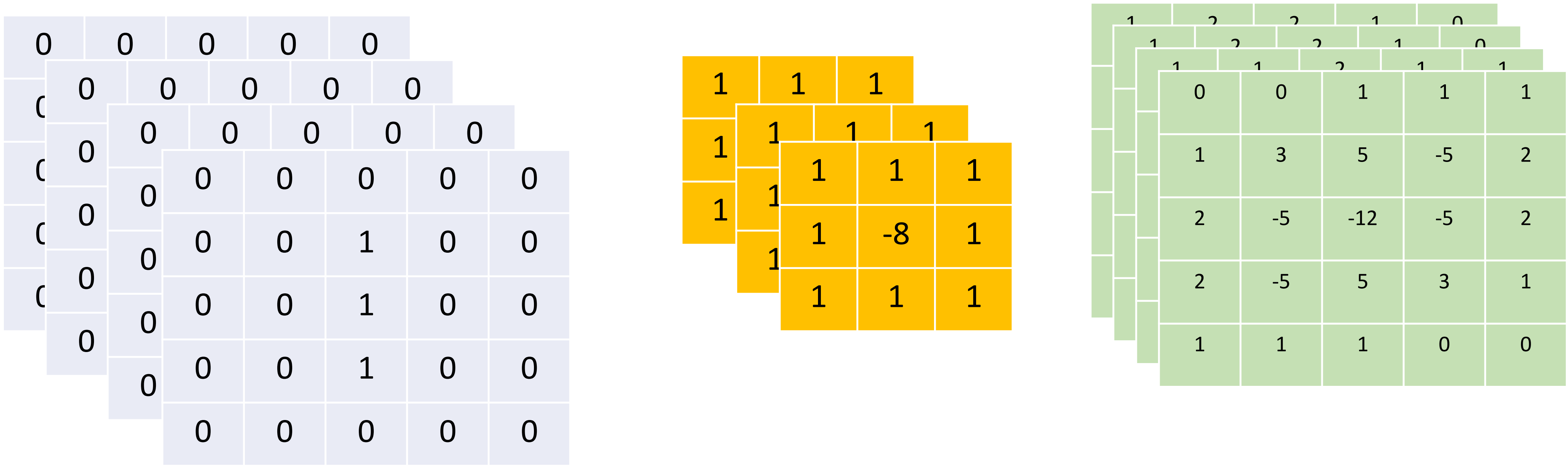
# 2D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

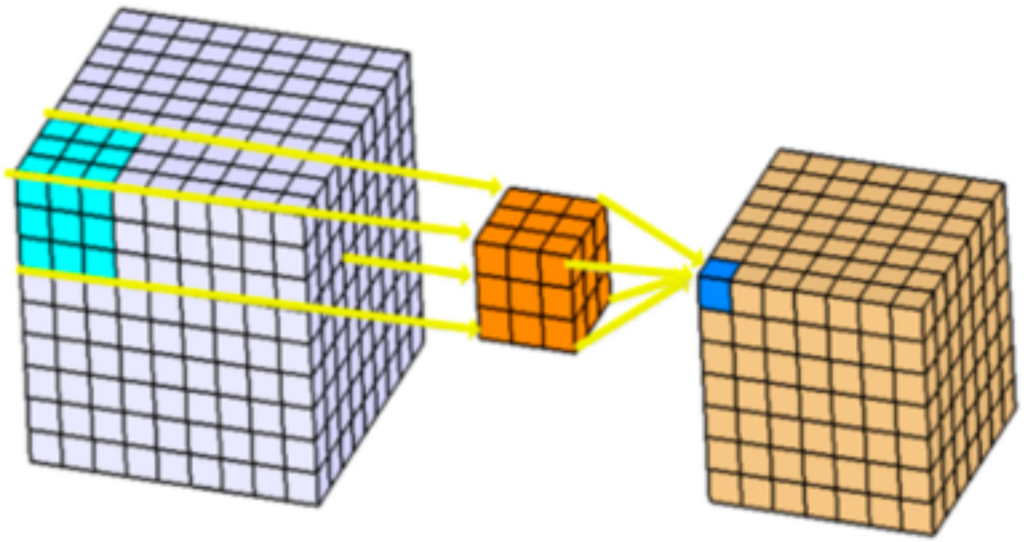
| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -8 | 1 |
| 1 | 1 | 1 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 2 | -7 | 1 |
| 1 | 2 | -6 | 2 | 1 |
| 1 | -7 | 2 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |

# 3D Convolution



# 3D Convolution



# 3D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -26 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

$$O_{i,j,k,c_j^{out}} = \sum_{h=1}^{c^{out}} \sum_{a,b,c,} w_{a,b,c}^{k} I_{i-a,j-b,k-c,h}$$

# 3D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -26 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| -21 | | |
| | | |
| | | |

$$O_{i,j,k,c_j^{out}} = \sum_{h=1}^{c^{out}} \sum_{a,b,c,} w_{a,b,c}^{k} I_{i-a,j-b,k-c,h}$$

# 3D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -26 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| -21 | 7 | |
| | | |
| | | |

$$O_{i,j,k,c_j^{out}} = \sum_{h=1}^{c^{out}} \sum_{a,b,c,} w^k_{a,b,c} I_{i-a,j-b,k-c,h}$$

# 3D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -26 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| -21 | 7 | 5 |
| | | |
| | | |

$$O_{i,j,k,c_j^{out}} = \sum_{h=1}^{c^{out}} \sum_{a,b,c,} w_{a,b,c}^{k} I_{i-a,j-b,k-c,h}$$

# 3D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -26 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| -21 | 7 | 5 |
| 7 | | |
| | | |

$$O_{i,j,k,c_j^{out}} = \sum_{h=1}^{c^{out}} \sum_{a,b,c,} w^{k}_{a,b,c} I_{i-a,j-b,k-c,h}$$

# 3D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -26 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| -21 | 7 | 5 |
| 7 | -18 | |
| | | |

$$O_{i,j,k,c_j^{out}} = \sum_{h=1}^{c^{out}} \sum_{a,b,c,} w^k_{a,b,c} I_{i-a,j-b,k-c,h}$$

# 3D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -26 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| -21 | 7 | 5 |
| 7 | -18 | 7 |
| | | |

$$O_{i,j,k,c_j^{out}} = \sum_{h=1}^{c^{out}} \sum_{a,b,c,} w_{a,b,c}^{k} I_{i-a,j-b,k-c,h}$$

# 3D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -26 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| -21 | 7 | 5 |
| 7 | -18 | 7 |
| 5 | 7 | -21 |

$$O_{i,j,k,c_j^{out}} = \sum_{h=1}^{c^{out}} \sum_{a,b,c,} w_{a,b,c}^{k} I_{i-a,j-b,k-c,h}$$

# 3D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | -26 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -26 | 1 |
| 1 | 1 | 1 |

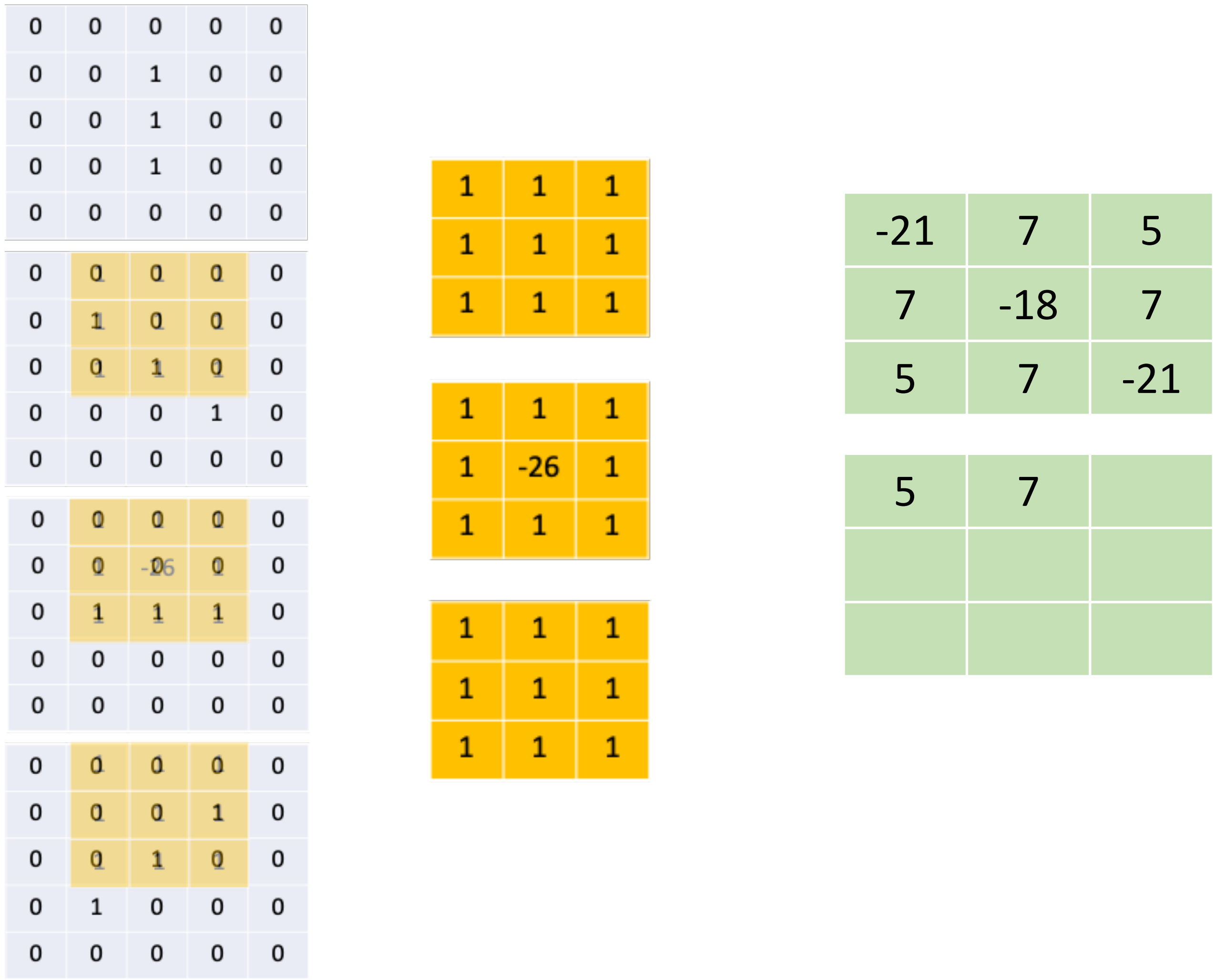
| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| -21 | 7 | 5 |
| 7 | -18 | 7 |
| 5 | 7 | -21 |

| | | |
|---|---|---|
| 5 | | |
| | | |
| | | |

$$O_{i,j,k,c_j^{out}} = \sum_{h=1}^{c^{out}} \sum_{a,b,c,} w^{k}_{a,b,c} I_{i-a,j-b,k-c,h}$$

# 3D Convolution




$$O_{i,j,k,c_j^{out}} = \sum_{h=1}^{c^{out}} \sum_{a,b,c,} w^{k}_{a,b,c} I_{i-a,j-b,k-c,h}$$

# 3D Convolution

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | -26 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

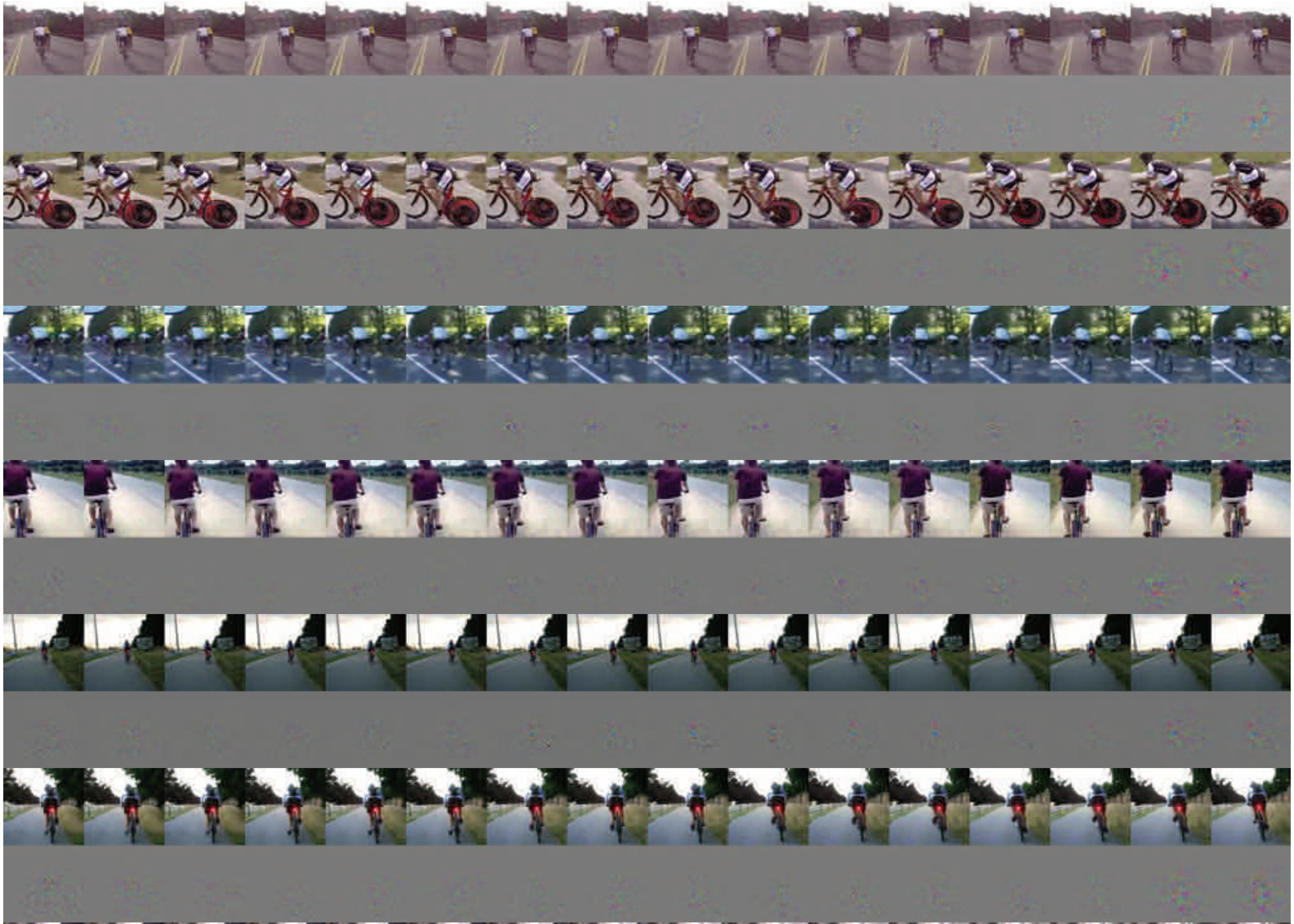
| | | |
|---|---|---|
| -21 | 7 | 5 |
| 7 | -18 | 7 |
| 5 | 7 | -21 |

| | | |
|---|---|---|
| 5 | 7 | 5 |
| -21 | -18 | -21 |
| 5 | 7 | 5 |

$$O_{i,j,k,c_j^{out}} = \sum_{h=1}^{c^{out}} \sum_{a,b,c,} w^{k}_{a,b,c} I_{i-a,j-b,k-c,h}$$

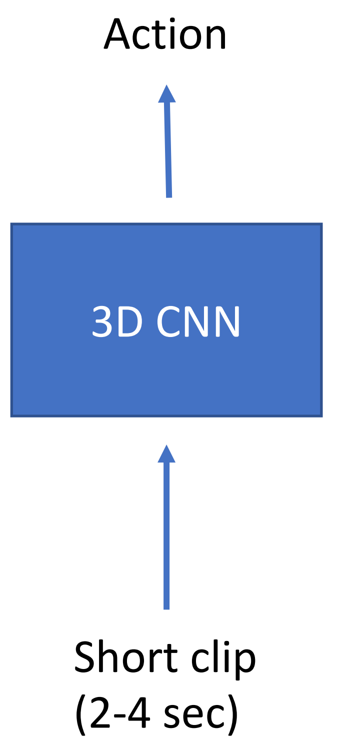
# Architetture basate su 3D Convolution

- C3D architecture
  - Simile a VGG
  - 8 convolution, 5 pool, 2 fully-connected layers – 3x3x3 convolution kernels, 2x2x2 pooling kernels



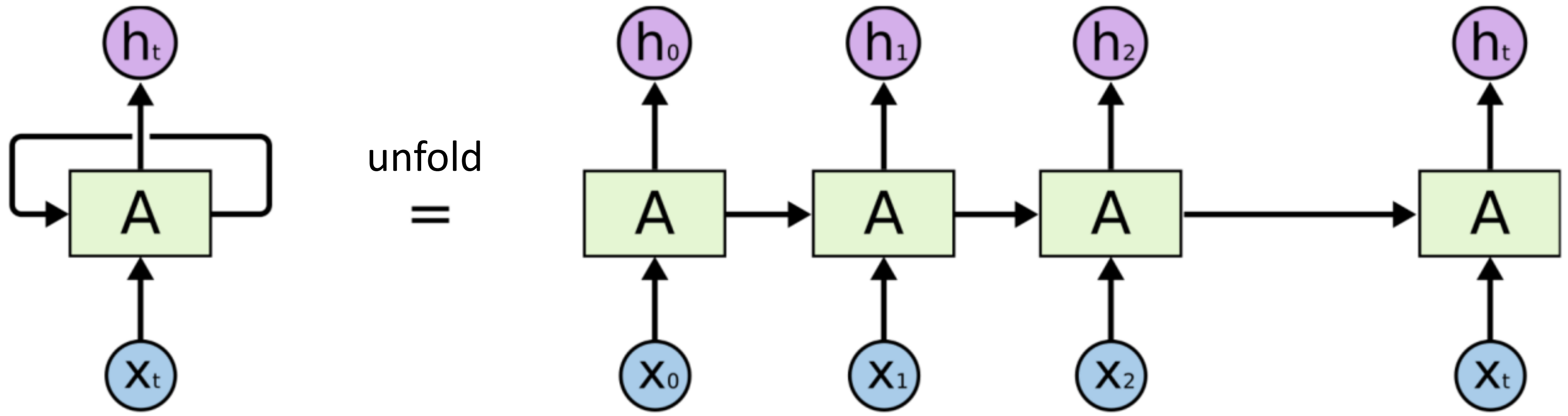
# 3D ConvNets

- Problema:
  - La lunghezza temporale dell'input dev'essere limitata
  - Come facciamo a recuperare relazioni long-term?



# Recurrent Neural Networks

- Un grafo con cicli
  - L’output di un perceptron al tempo $t$ è concatenato all’input al tempo $t+1$



# Perché le RNN?

- Possono modellare conoscenza su sequenze
- Effetto memoria

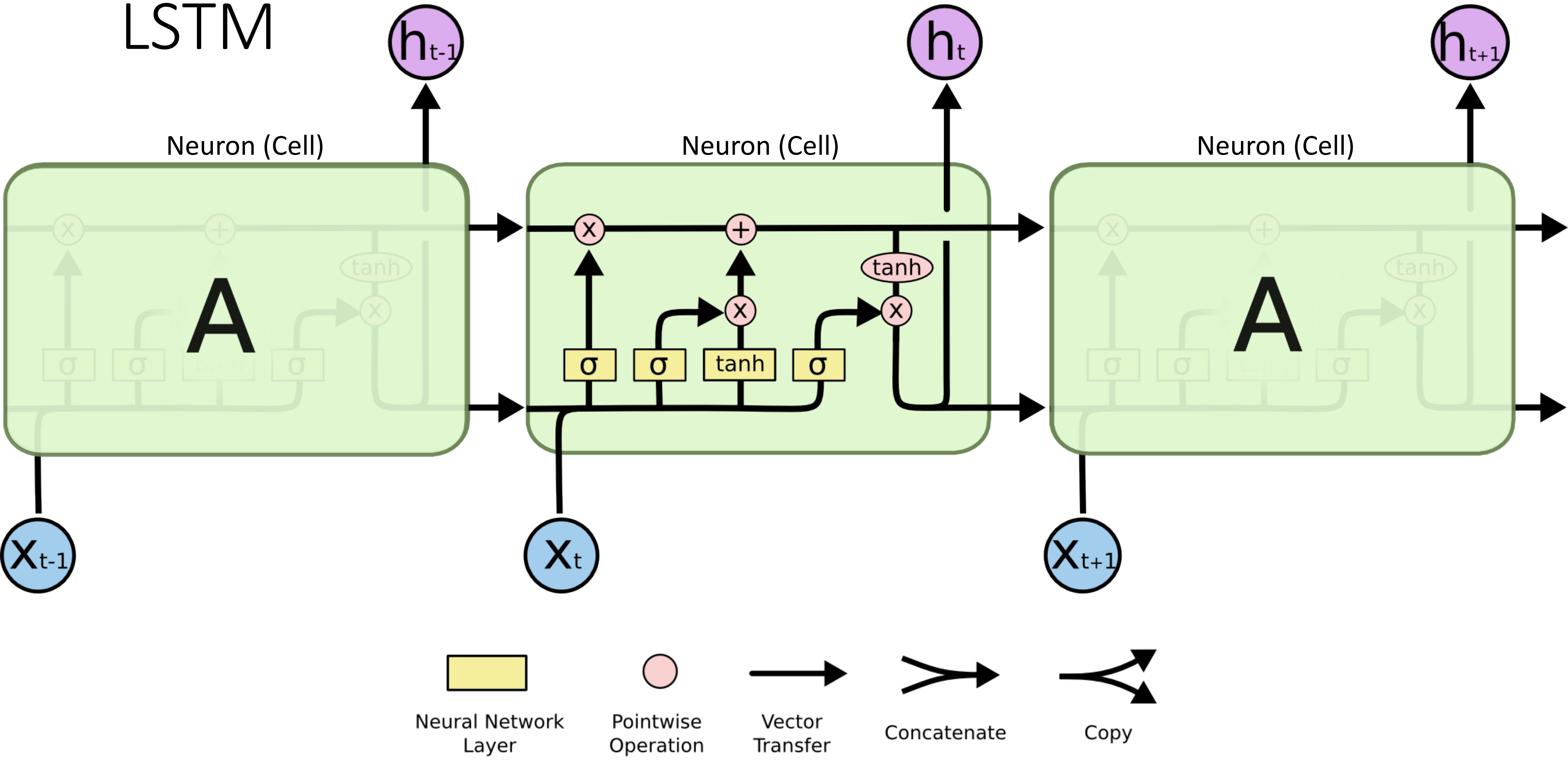
# Backpropagation Through Time

- L’addestramento avviene tramite unfolding
  - I loop corrispondono ad una rete very deep
  - Variante: l’unfolding viene tagliato ad una certa distanza
    - Backpropagation through time (BPTT)

# Long Short Term Memory

- Un tipo speciale di RNN, capace di gestire le relazioni long-term tramite gating
  - Controllo sul vanishing gradien
  - L’informazione nel lungo period viene propagate in avanti

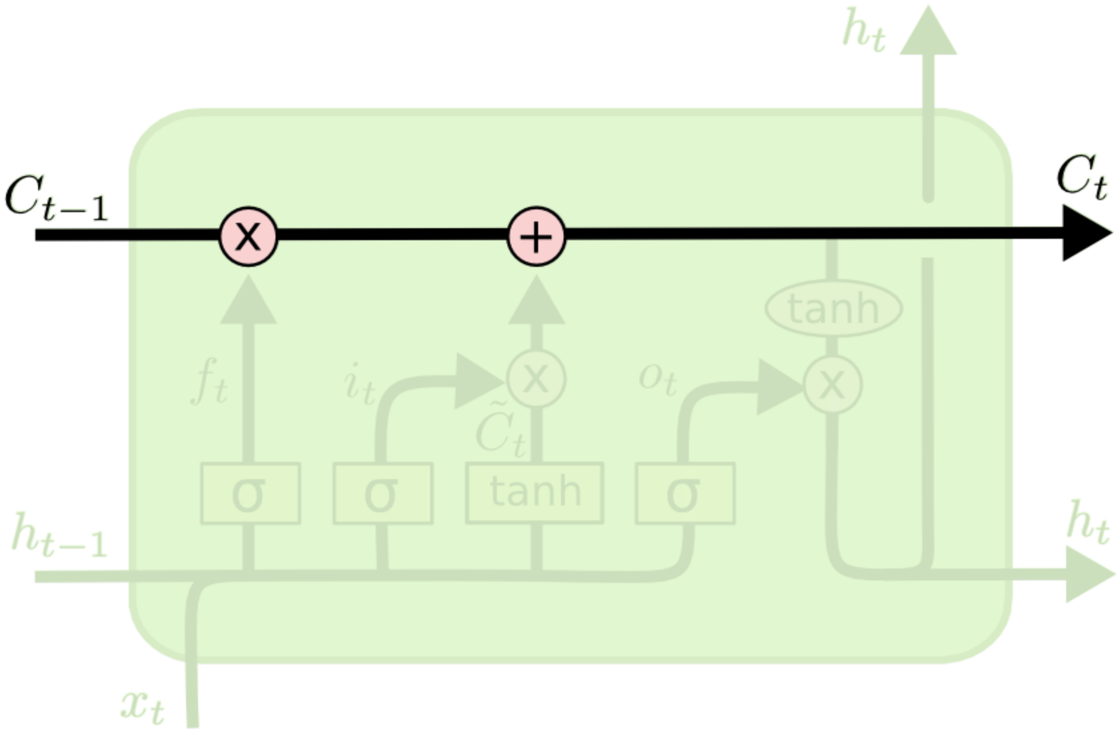
# LSTM



$h_{t-1}$
$h_t$
$h_{t+1}$
Neuron (Cell)
Neuron (Cell)
Neuron (Cell)
A
A
σ
σ
tanh
σ
tanh
$x_{t-1}$
$x_t$
$x_{t+1}$
Neural Network Layer
Pointwise Operation
Vector Transfer
Concatenate
Copy

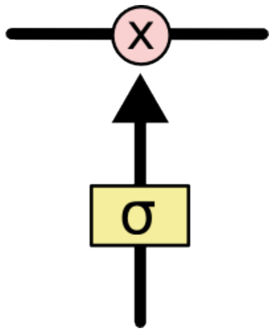
# LSTM

- La cella ha uno stato interno
  - fluisce lungo la catena con interazioni limitate



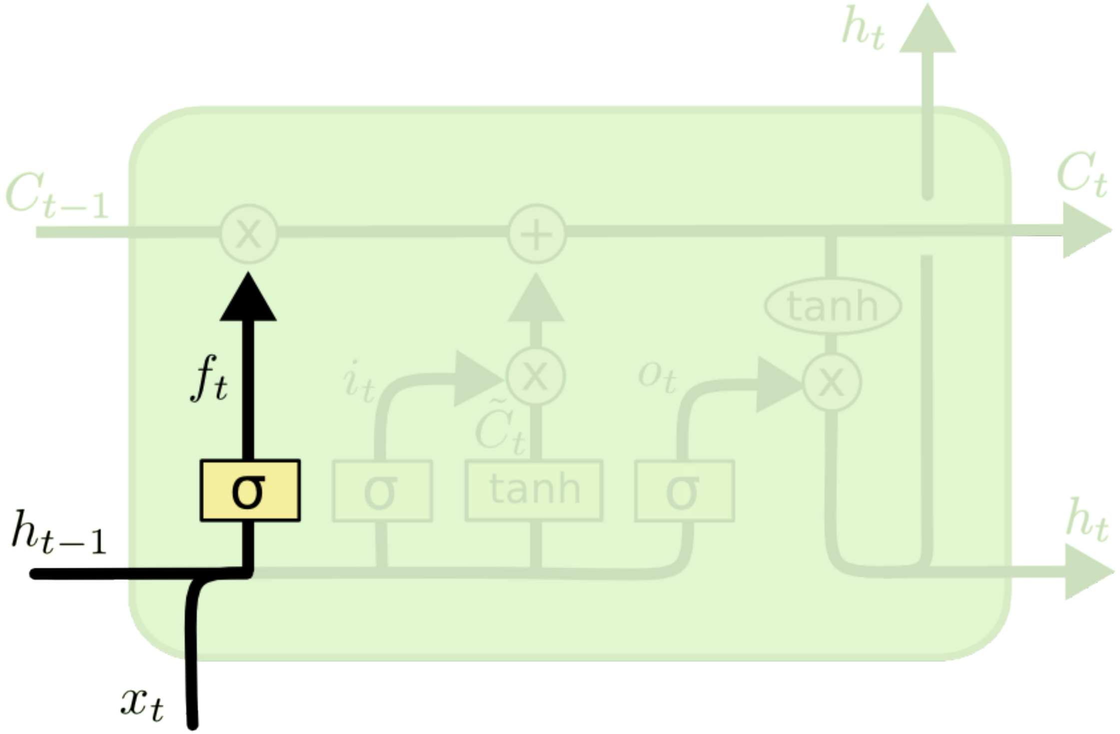
# LSTM

- Il meccanismo di gating permette di rimuovere o aggiungere informazione allo stato



# LSTM

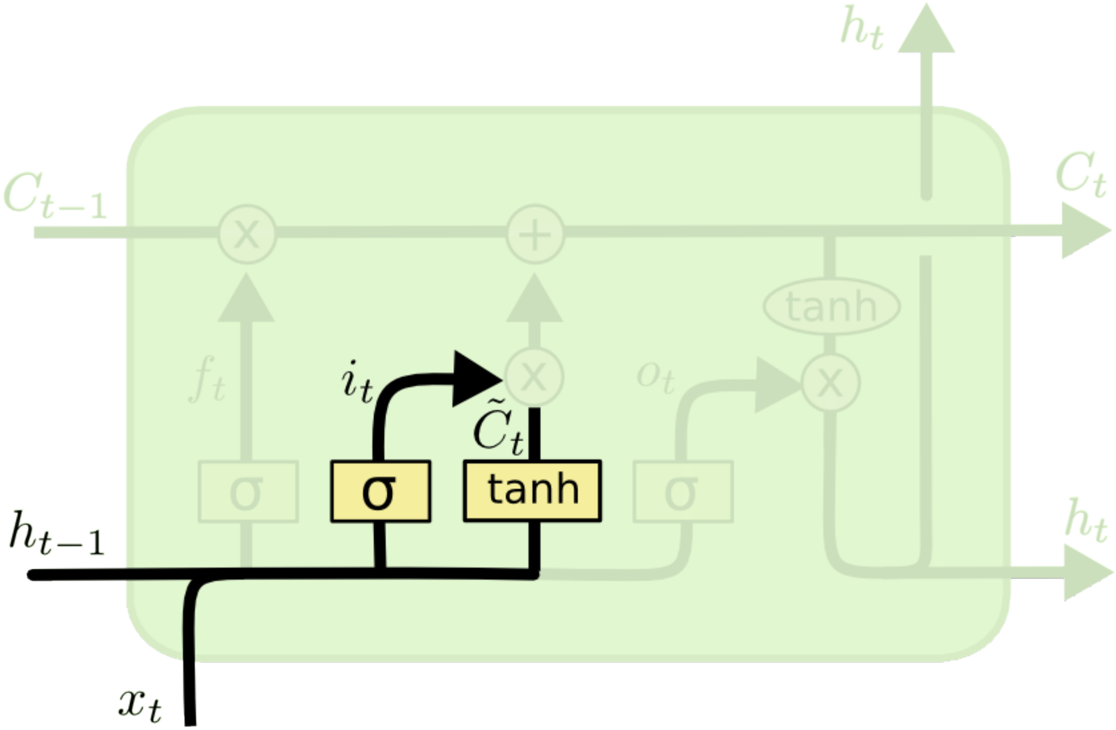
- forget gate
  - Determina quanta informazione precedente considerare




$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

# LSTM

- Input gate
  - Quanto il nuovo input può influenzare la storia corrente

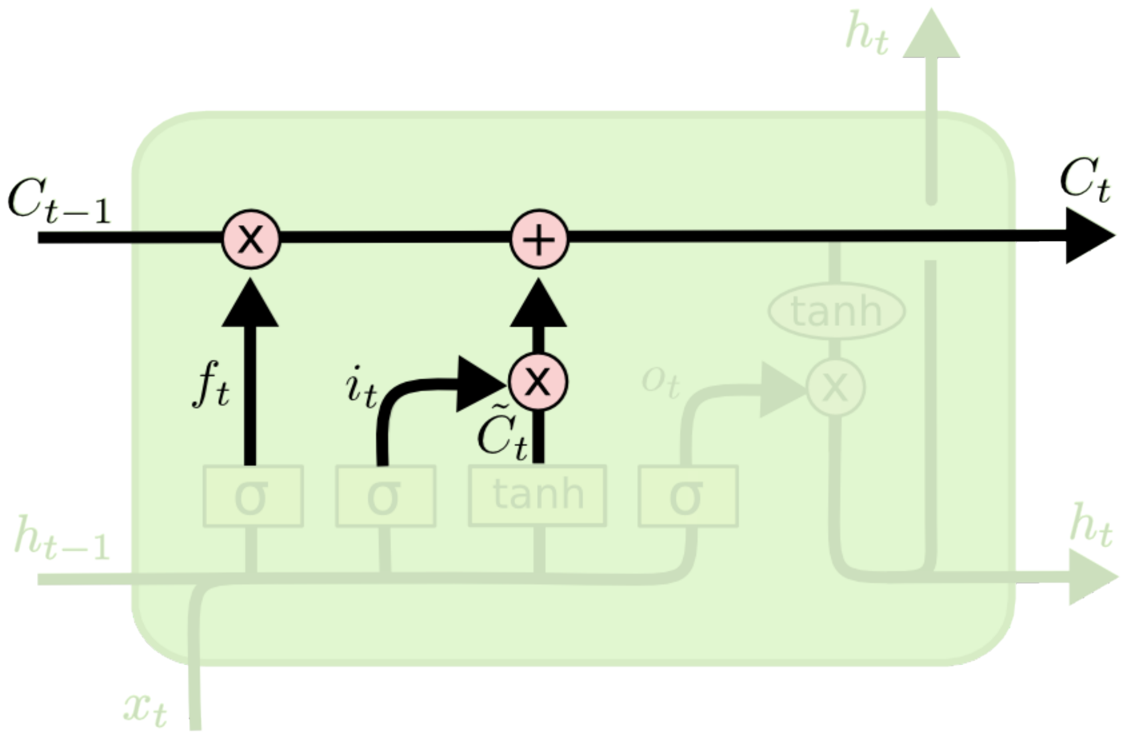



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

# LSTM

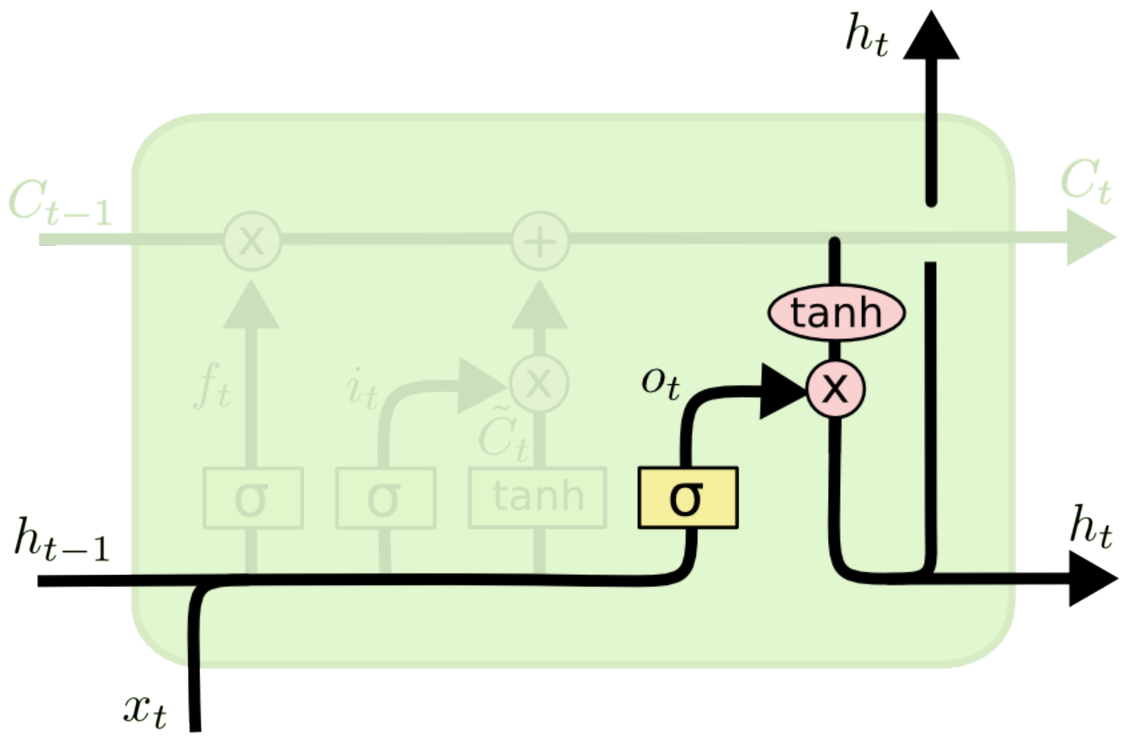
- Update state
  - Gestione del gradiente evanescente…




$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

# LSTM

- Output e nuovo stato

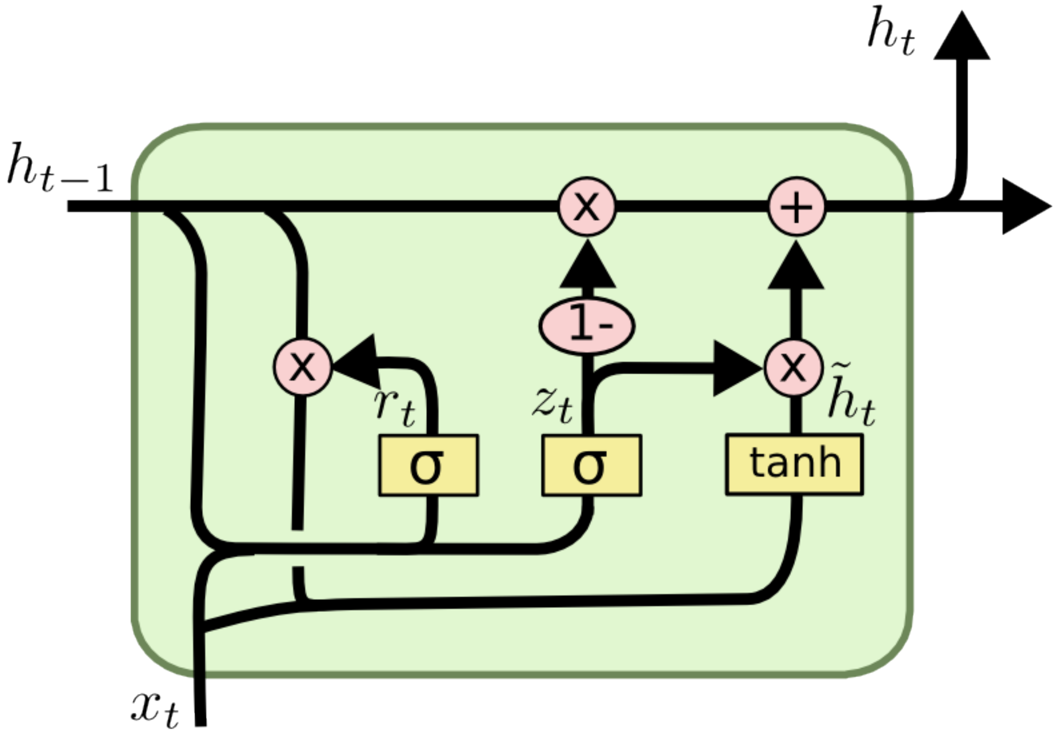



$$o_t = \sigma\left(W_o\ [h_{t-1}, x_t]\ +\ b_o\right)$$

$$h_t = o_t * \tanh\left(C_t\right)$$

# Varianti

- Gated Recurrent Unit (GRU)
  - Più rilavanza alla storia, controllando i gate




$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

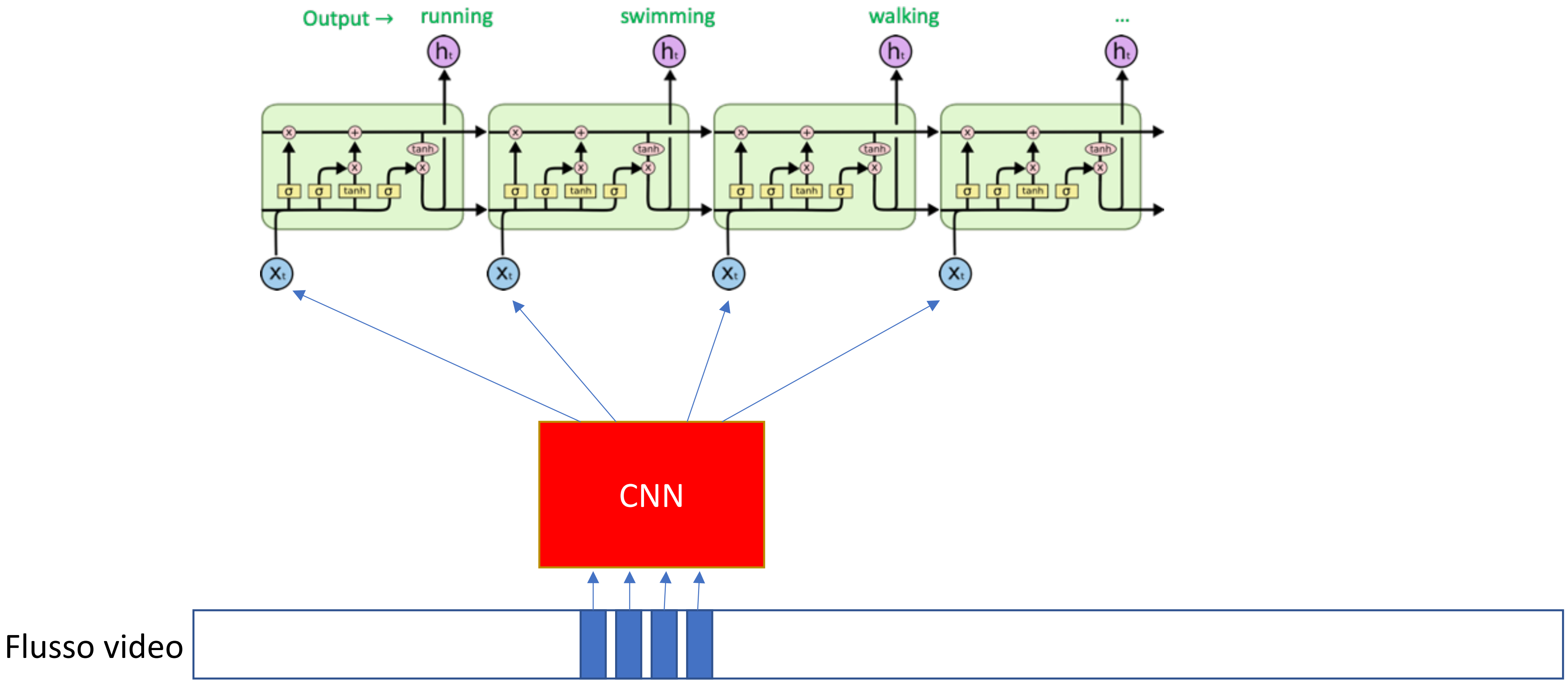
$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

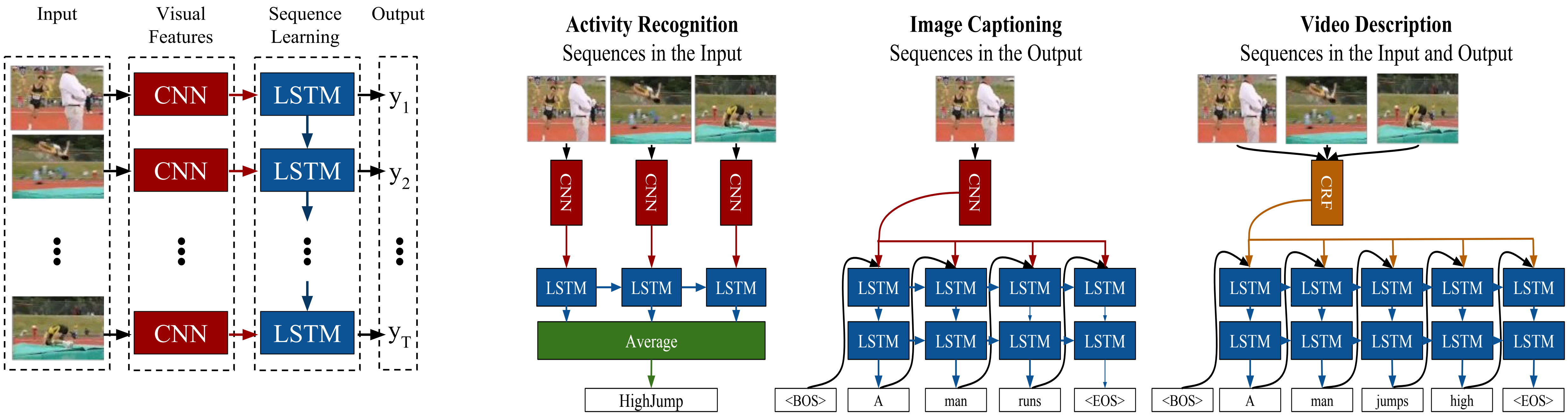
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

# RNN e action recognition

- Input: encoding dei frames
  - Long-term Recurrent Con- volutional Networks (LRCNs)

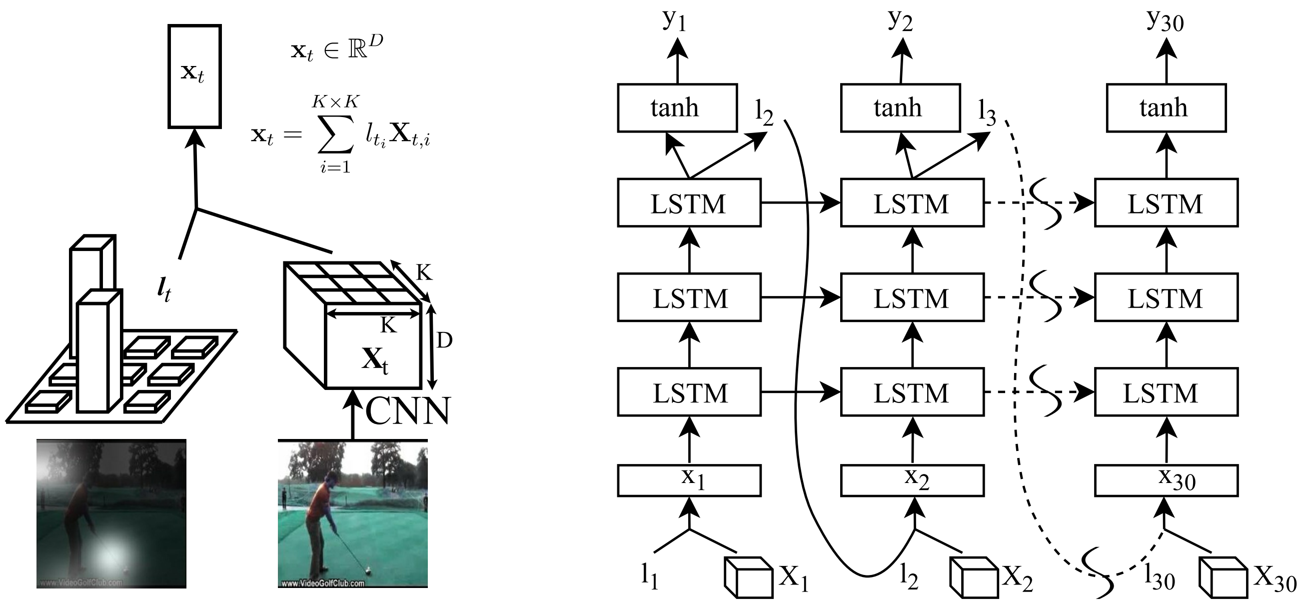


# Combinazioni flessibili



# RNN + Attention

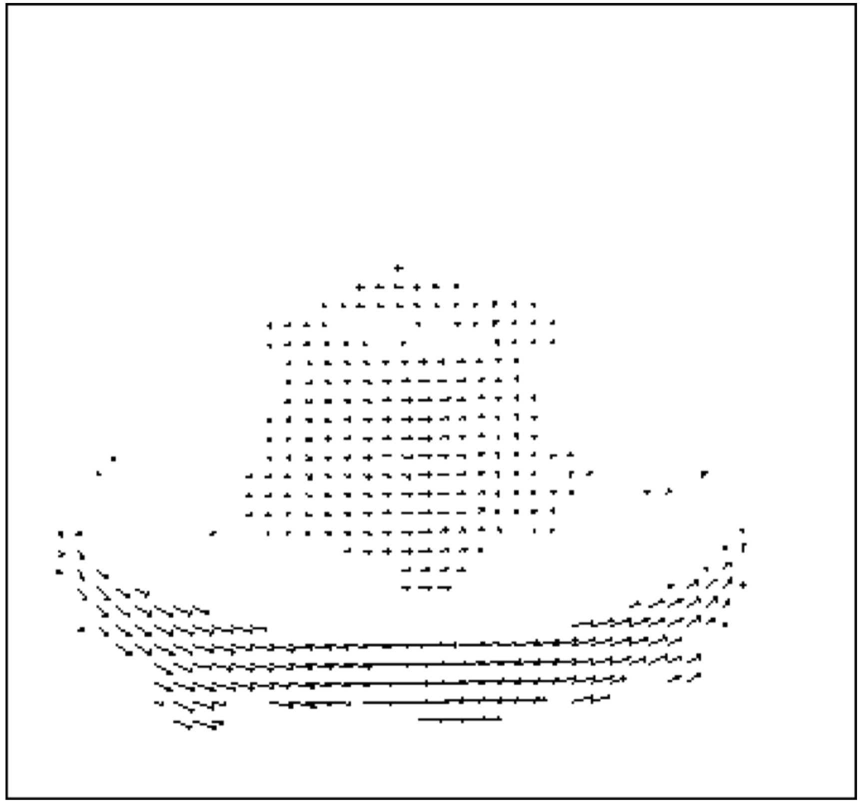
- I pesi sulle features sono distribuiti in base allo storico



# Optical Flow

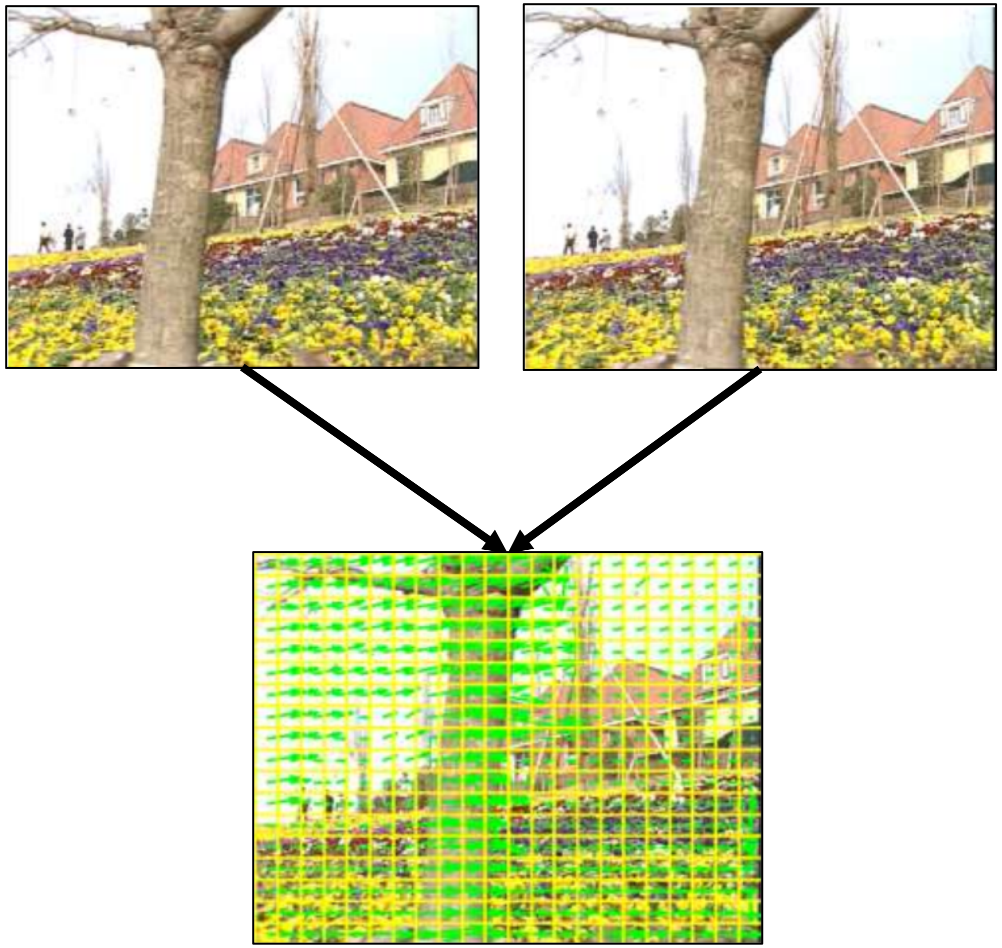
- Idea:
  - Catturiamo la nozione di movimento in termini vettoriali
    - Lo spostamento dei pixel all'interno dell'immagine col tempo

# Optical Flow: i pixel si muovono



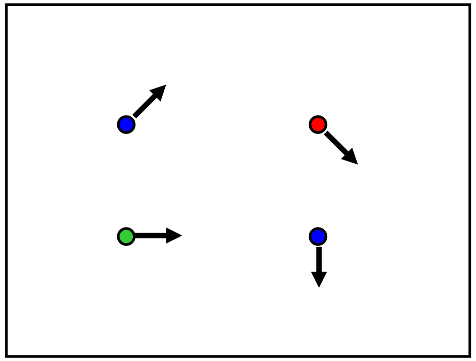
# A cosa serve?

**Motion Estimation**

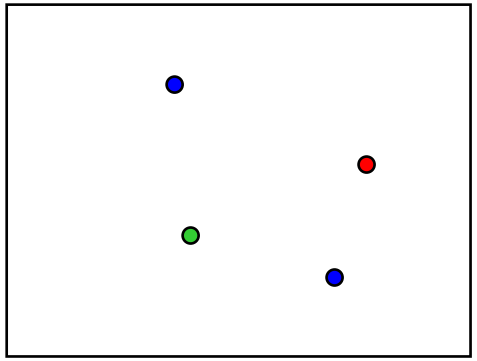


**Object Tracking**

# Matematica del flusso ottico



$H(x, y)$

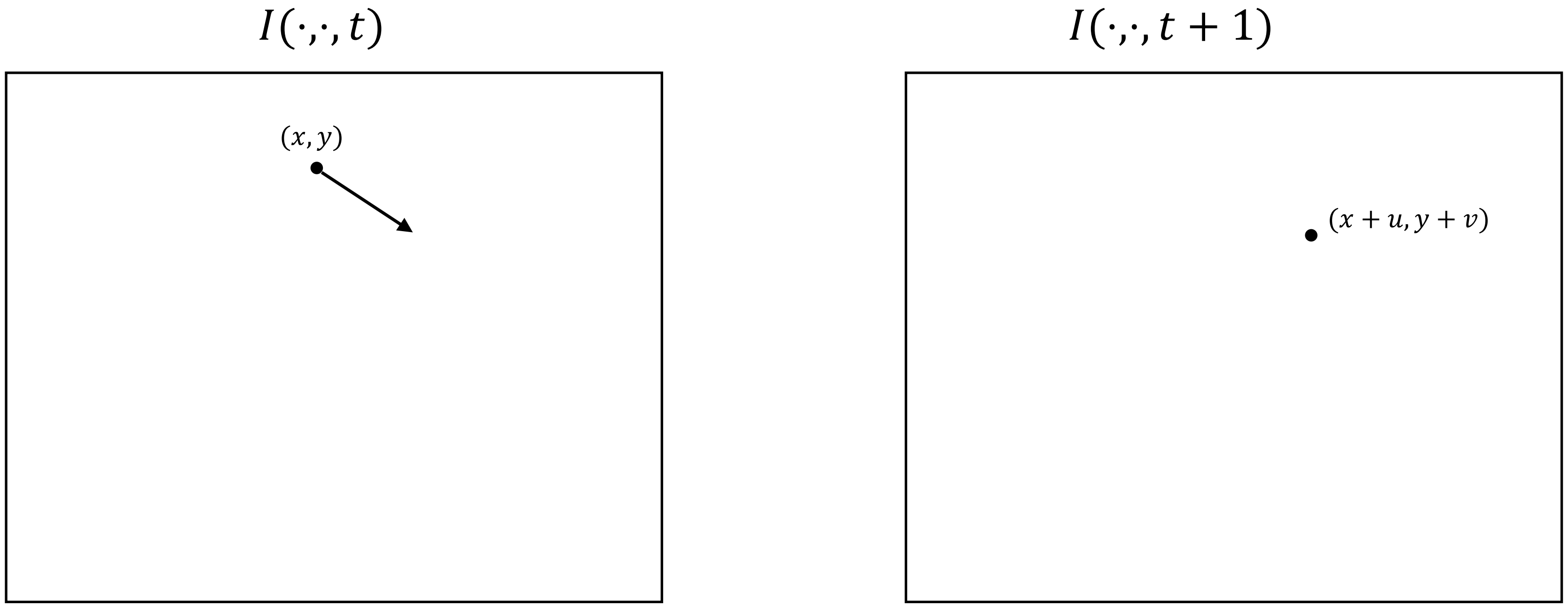


$I(x, y)$

- Corrispondenza di pixel
  - Dato un pixel in H, trova il pixel corrispondente in I
- Assunzioni
  - Le intensità non cambiano
  - I punti si spostano di poco

# Idea

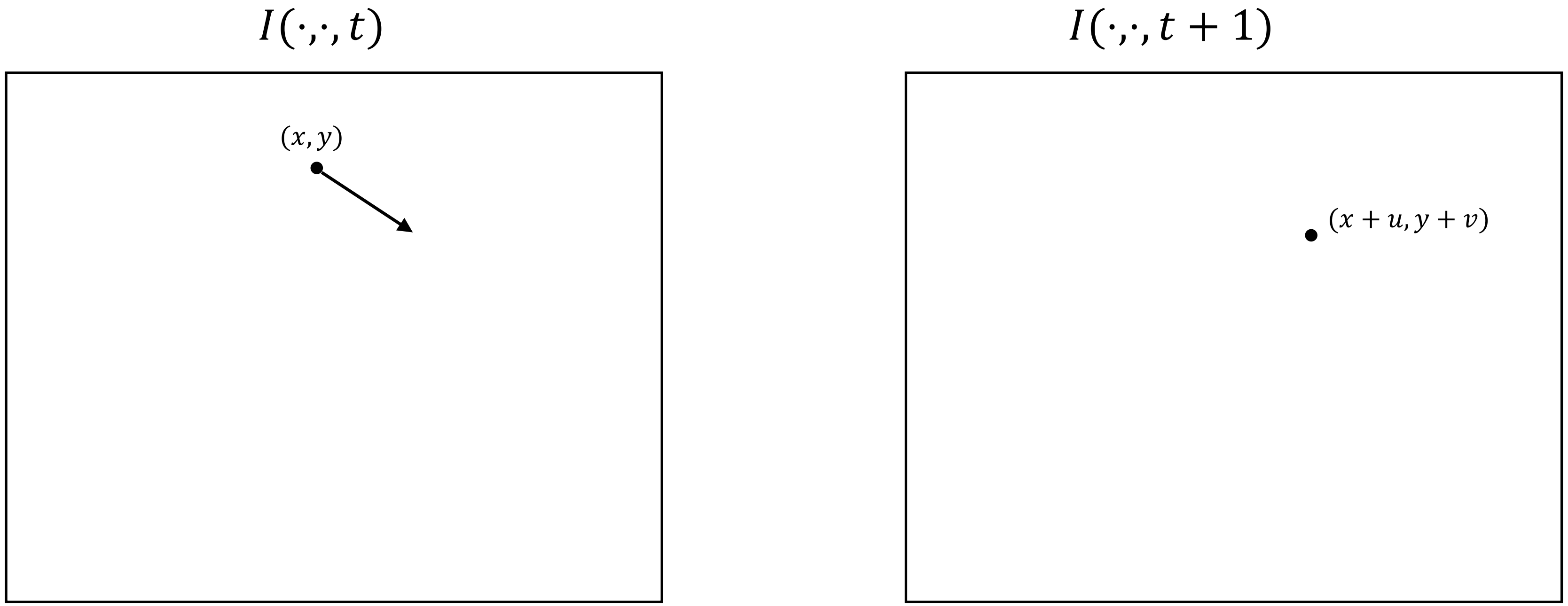
- $(u, v)$ vettore di spostamento



# Idea

- $(u, v)$ vettore di spostamento

$$I(x + u, y + v, t) = I(x, y, t + 1)$$



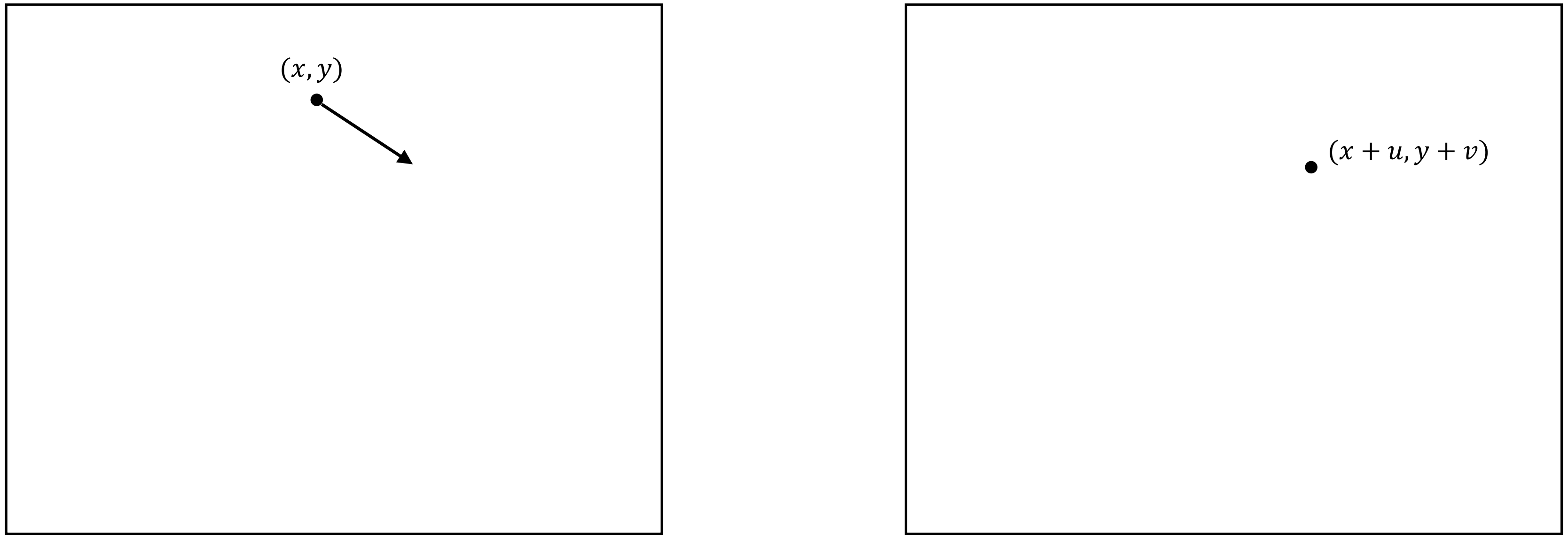
# Idea

- $(u, v)$ vettore di spostamento
  - Approssimando in serie di Taylor al primo ordine

$$f(x+u, y+v) \approx f(x,y) + u\frac{\partial f(x,y)}{\partial x} + v\frac{\partial f(x,y)}{\partial v}$$

$$I(x+u, y+v, t) = I(x, y, t+1)$$

$I(\cdot,\cdot,t)$ $I(\cdot,\cdot,t+1)$



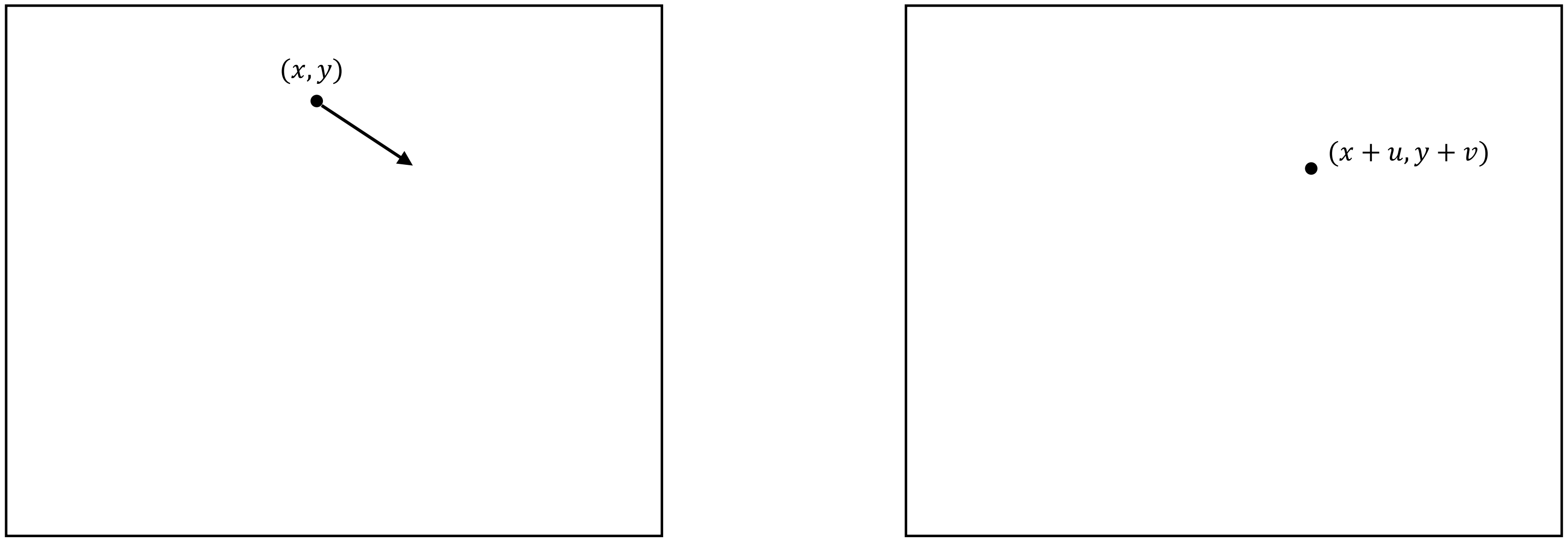
# Idea

- $(u, v)$ vettore di spostamento
  - Approssimando in serie di Taylor al primo ordine

$$f(x+u, y+v) \approx f(x,y) + u\frac{\partial f(x,y)}{\partial x} + v\frac{\partial f(x,y)}{\partial y}$$

$$I(x,y,t) + u\frac{\partial I(x,y,t)}{\partial x} + v\frac{\partial I(x,y,t)}{\partial y} \approx I(x,y,t+1)$$

$I(\cdot,\cdot,t)$ $\qquad\qquad$ $I(\cdot,\cdot,t+1)$

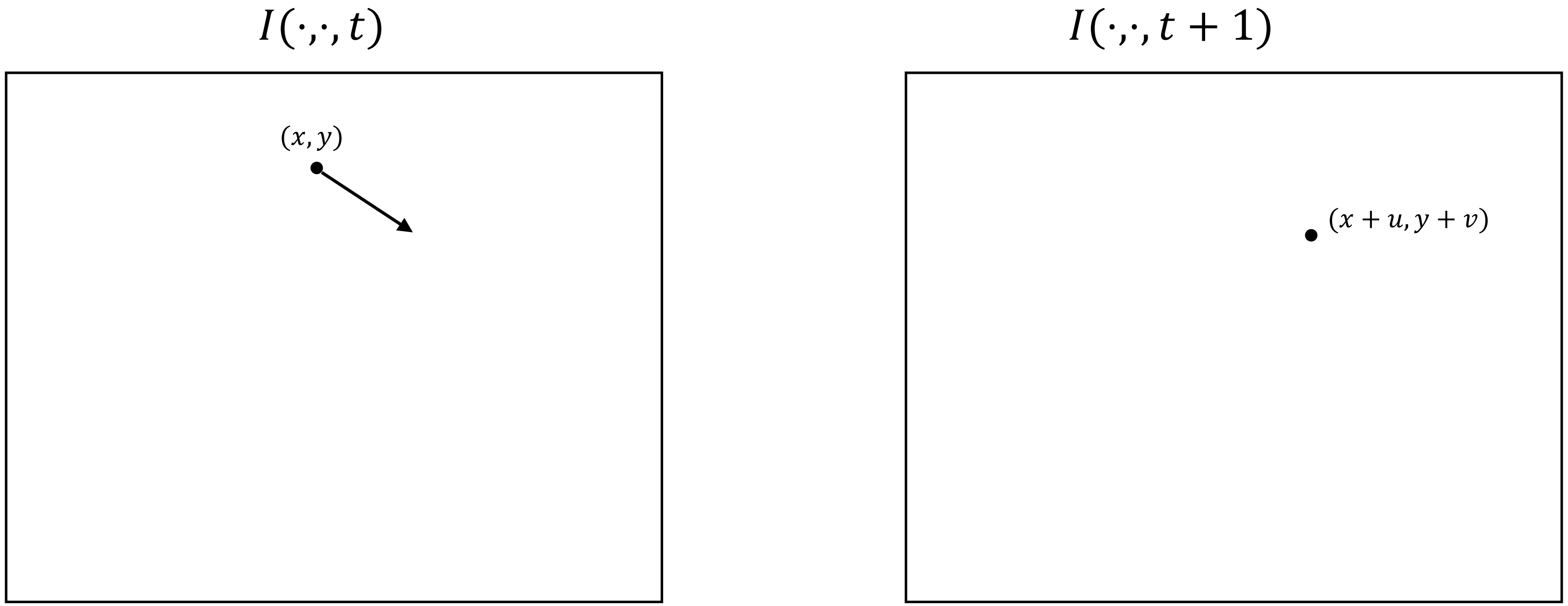


# Idea

- $(u, v)$ vettore di spostamento
  - Approssimando in serie di Taylor al primo ordine

$$f(x+u, y+v) \approx f(x,y) + u\frac{\partial f(x,y)}{\partial x} + v\frac{\partial f(x,y)}{\partial y}$$

$$u\frac{\partial I(x,y,t)}{\partial x} + v\frac{\partial I(x,y,t)}{\partial y} \approx I(x,y,t+1) - I(x,y,t)$$

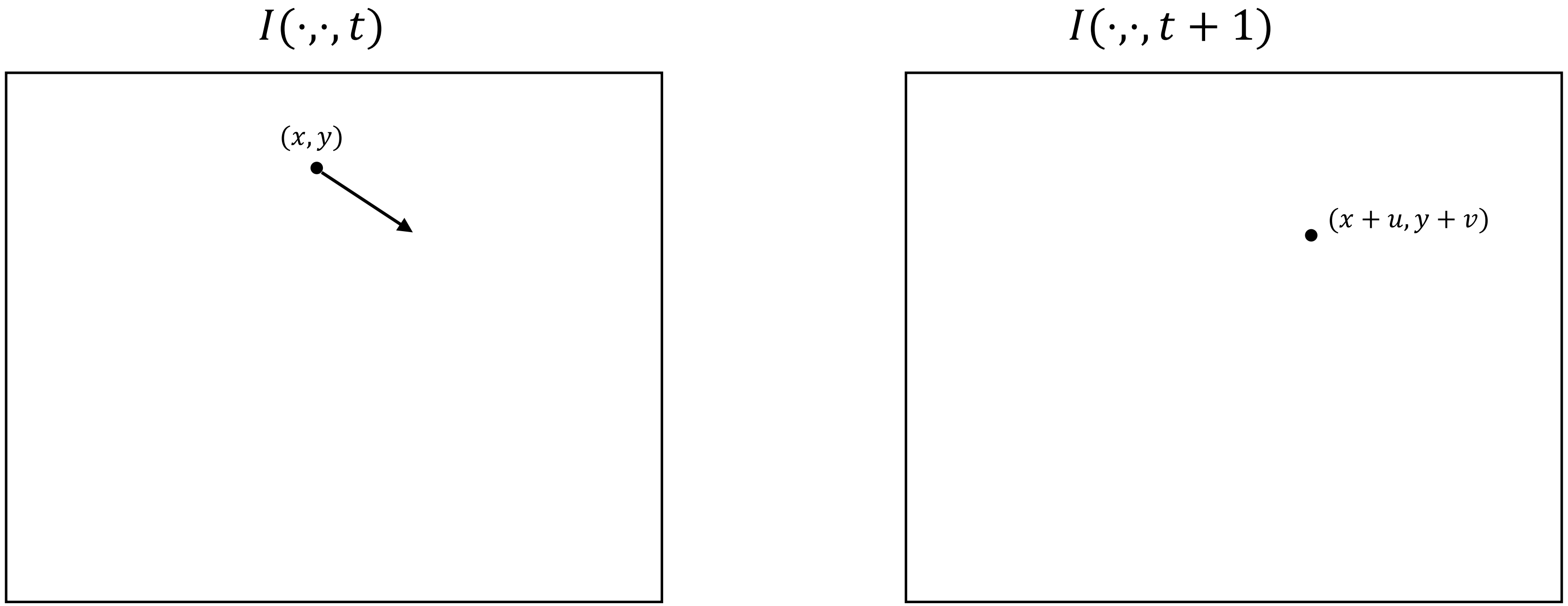


# Idea

- $(u, v)$ vettore di spostamento
  - Approssimando in serie di Taylor al primo ordine

$$f(x+u, y+v) \approx f(x,y) + u\frac{\partial f(x,y)}{\partial x} + v\frac{\partial f(x,y)}{\partial y}$$

$$uI_x(x,y,t) + \ vI_y(x,y,t) \approx I_t(x,y)$$



# Optical Flow

- Risolvendo l'equazione troviamo $(u, v)$

$$uI_x(x, y, t) + \ vI_y(x, y, t) \approx I_t(x, y)$$

- Sovraspecificato

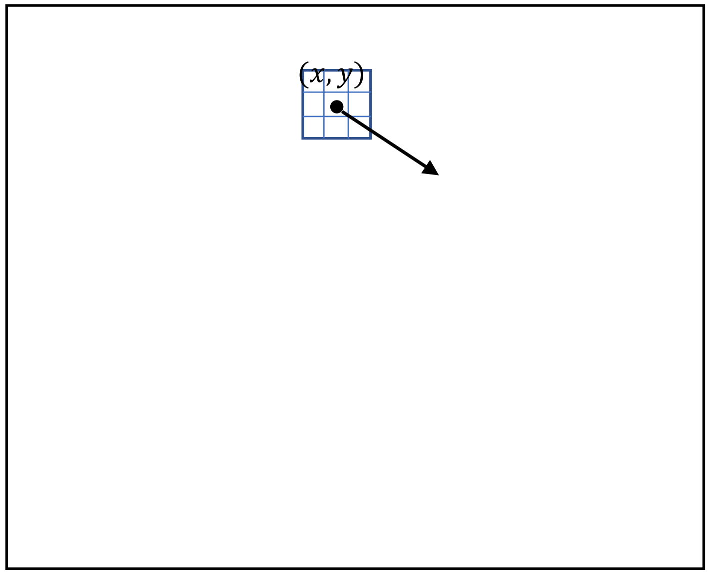
# Optical Flow

- Risolvendo l'equazione troviamo $(u, v)$

$$uI_x(x, y, t) + \ vI_y(x, y, t) \approx I_t(x, y)$$

- Idea: prendiamo un intorno di $(x, y)$

$I(\cdot,\cdot,t)$

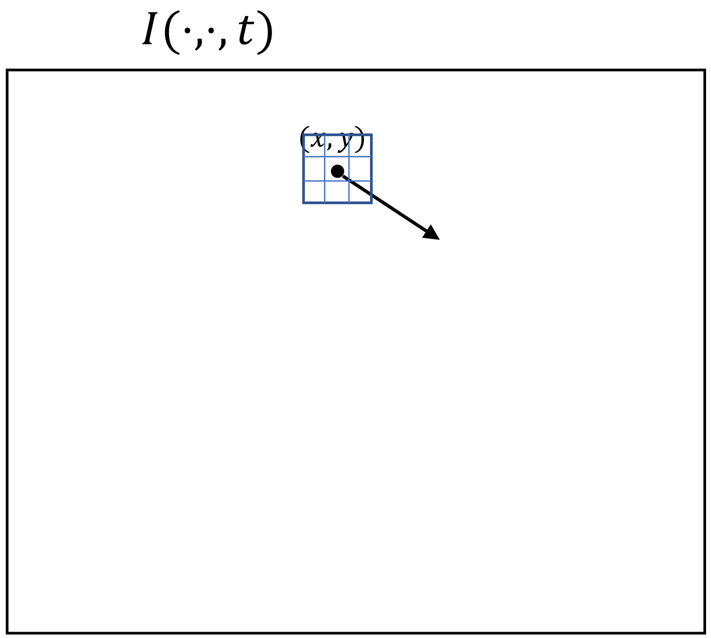


# Optical Flow

- Risolvendo l’equazione troviamo $(u, v)$

$$uI_x(x, y, t) + \ vI_y(x, y, t) \approx I_t(x, y)$$

- Idea: prendiamo un intorno di $(x, y)$
  - Lo spostamento è simile localmente



# Optical Flow

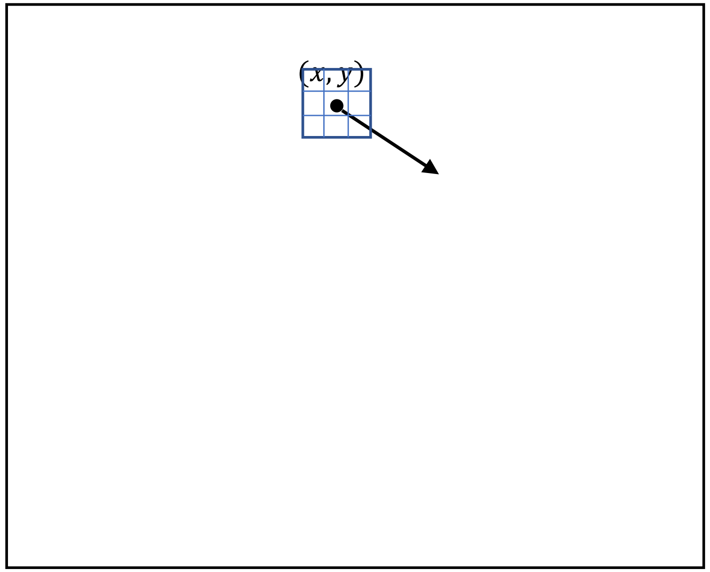
- Risolvendo l'equazione troviamo $(u, v)$

$$uI_x(x, y, t) + \; vI_y(x, y, t) \approx I_t(x, y)$$

- Idea: prendiamo un intorno di $(x, y)$
  - Lo spostamento è simile localmente

$$\begin{bmatrix} I_x(x_0, y_0) & I_y(x_0, y_0) \\ \dots & \dots \\ I_x(x_n, y_n) & I_y(x_n, y_n) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} I_t(x_0, y_0) \\ \dots \\ I_y(x_n, y_n) \end{bmatrix}$$

$I(\cdot,\cdot,t)$



# Optical Flow

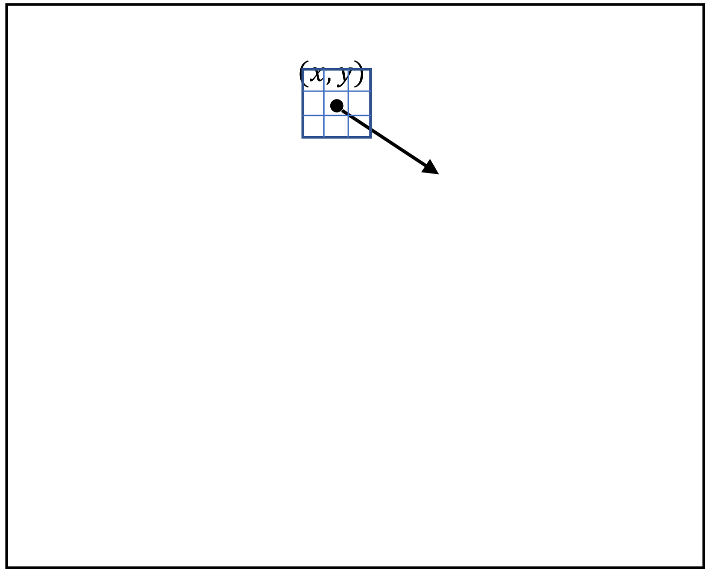
- Risolvendo l'equazione troviamo $(u, v)$

$$uI_x(x, y, t) + \ vI_y(x, y, t) \approx I_t(x, y)$$

- Idea: prendiamo un intorno di $(x, y)$
  - Lo spostamento è simile localmente

$$\underbrace{\begin{bmatrix} I_x(x_0, y_0) & I_y(x_0, y_0) \\ \ldots & \ldots \\ I_x(x_n, y_n) & I_y(x_n, y_n) \end{bmatrix}}_{A} \begin{bmatrix} u \\ v \end{bmatrix} = \underbrace{\begin{bmatrix} I_t(x_0, y_0) \\ \ldots \\ I_y(x_n, y_n) \end{bmatrix}}_{b}$$

$I(\cdot,\cdot,t)$



# Optical Flow

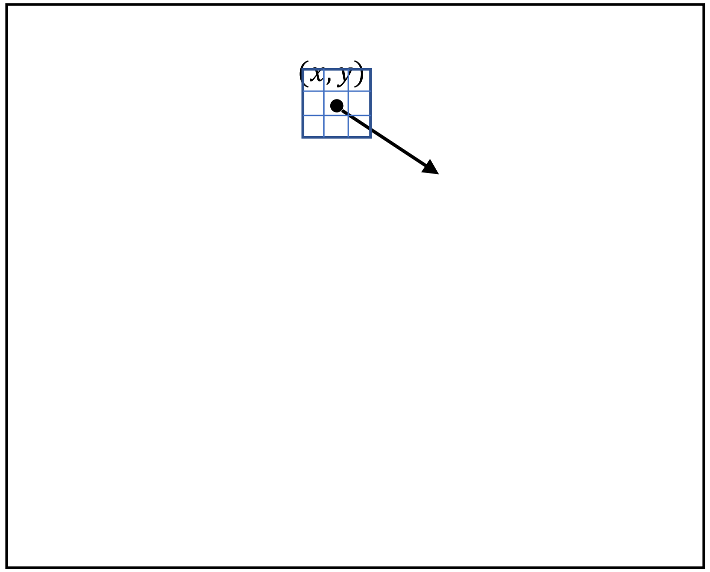
- Risolvendo l'equazione troviamo $(u, v)$

$$uI_x(x, y, t) + \ vI_y(x, y, t) \approx I_t(x, y)$$

- Idea: prendiamo un intorno di $(x, y)$
  - Lo spostamento è simile localmente

$$A\begin{bmatrix} u \\ v \end{bmatrix} = b$$

$I(\cdot, \cdot, t)$



# Optical Flow

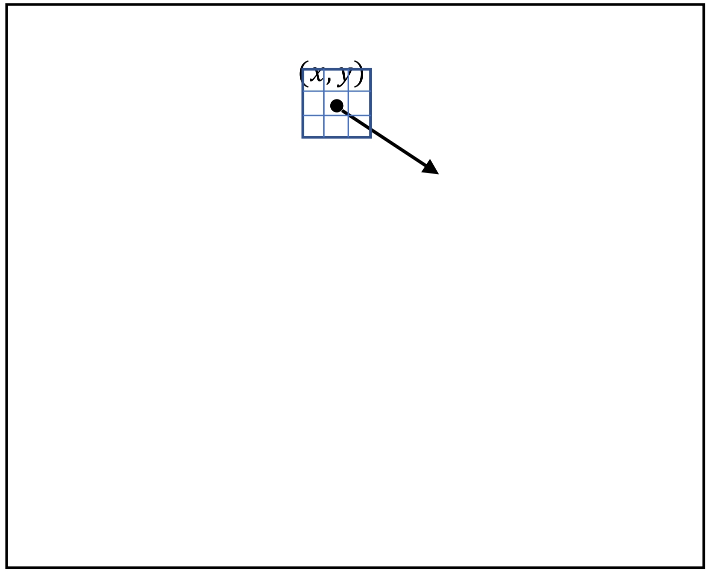
- Risolvendo l'equazione troviamo $(u, v)$

$$uI_x(x,y,t) + \; vI_y(x,y,t) \approx I_t(x,y)$$

- Idea: prendiamo un intorno di $(x, y)$
  - Lo spostamento è simile localmente

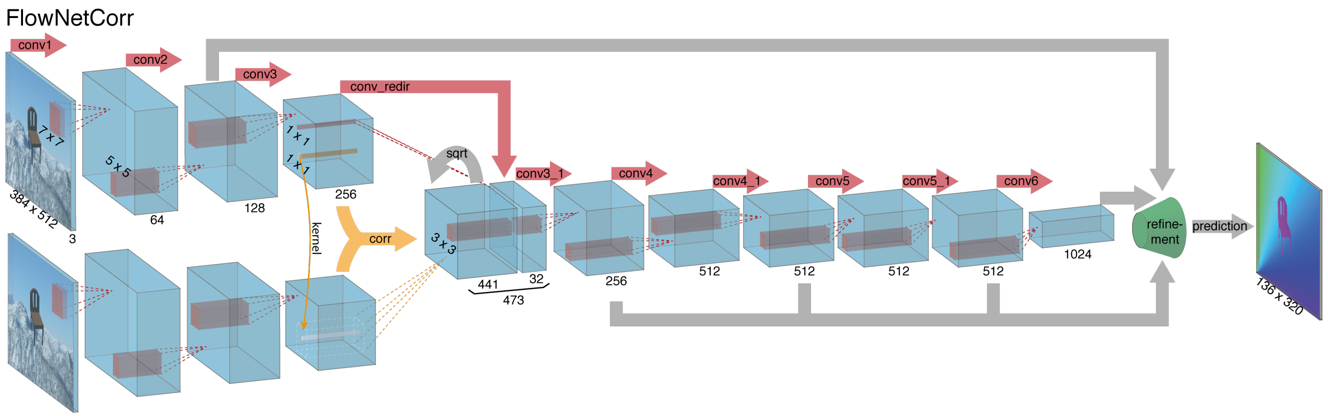
$$\begin{bmatrix} u \\ v \end{bmatrix} = (A^T A)^{-1} A^T b$$

$I(\cdot,\cdot,t)$



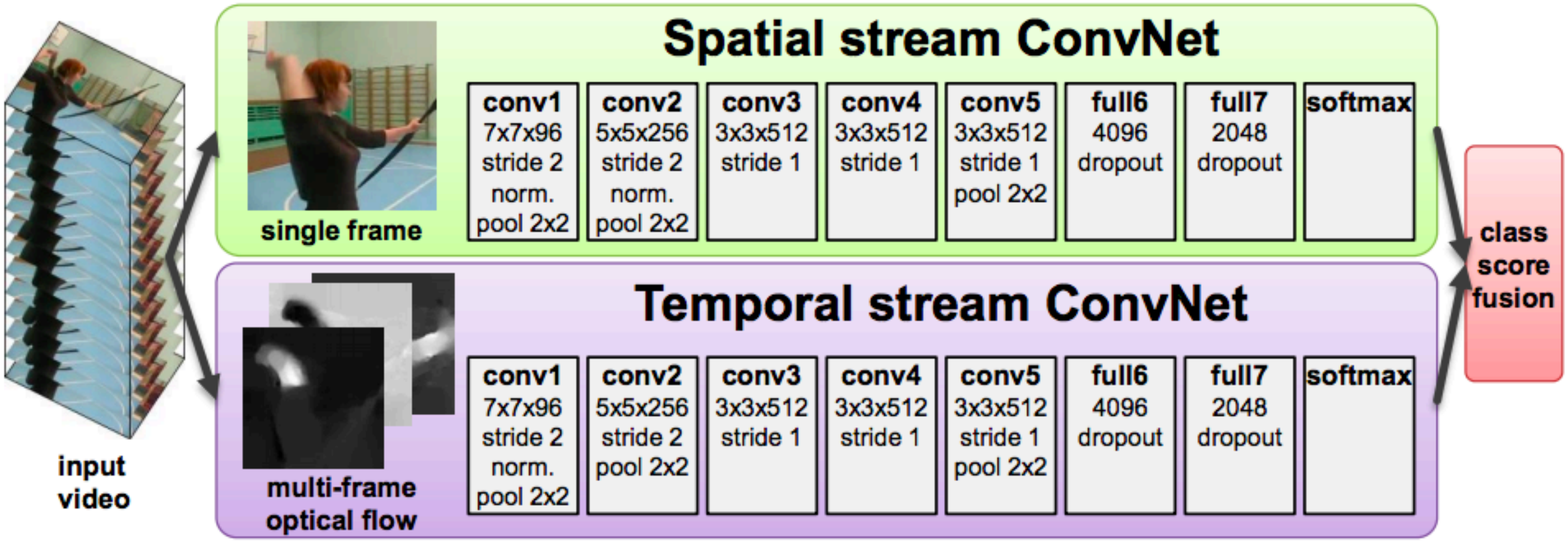
# Estensioni

- Apertura
- Pyramids
- Calcolo denso
  - Approssimazione al secondo ordine
  - Trasformazioni affini
- Learning optical Flow
  - FlowNet



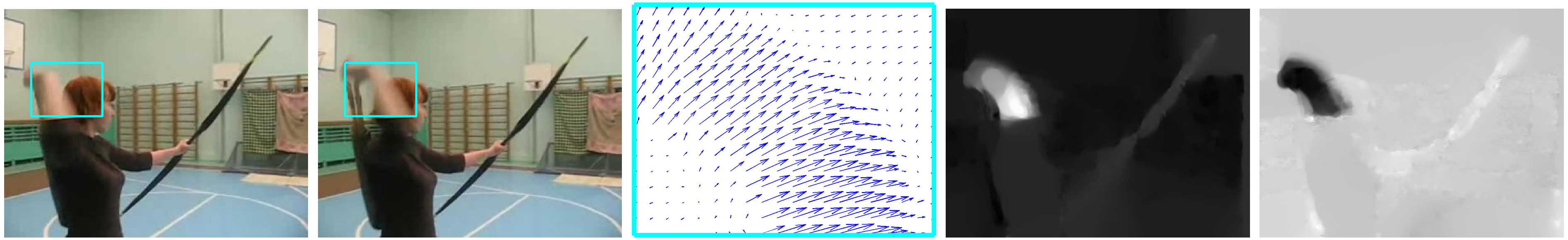
# Perché ci interessa il flusso ottico?

- Two-Stream Networks



# Optical Flow ConvNets

- Dati L frames consecutivi, lo stream temporale consiste di 2L canali di input
  - $I(x, y, c) = d_x(x, y)$
  - $I(x, y, c + 1) = d_y(x, y)$
  - $(d_x(x, y), d\_y(x, y))$ rappresenta il displacement vector



# Riassunto: Una pletora di alternative