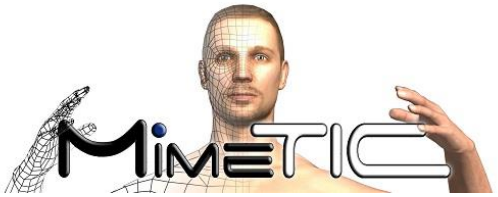
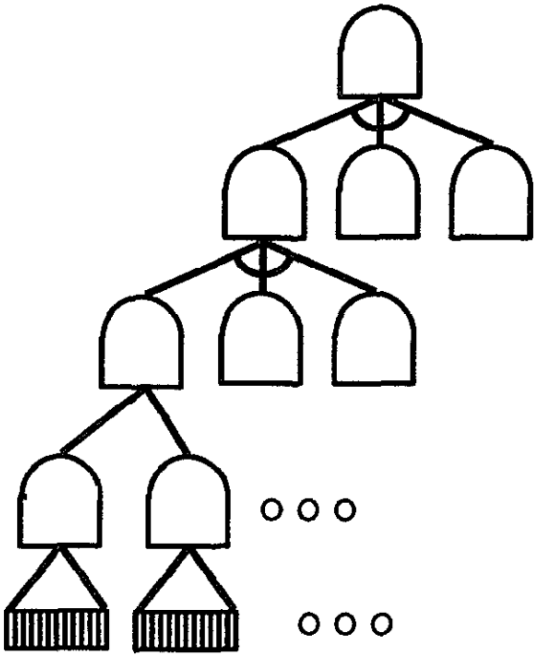
Towards Data-driven Virtual Cinematography

Christophe Lino



Insights from real film techniques

- « Film Grammar »



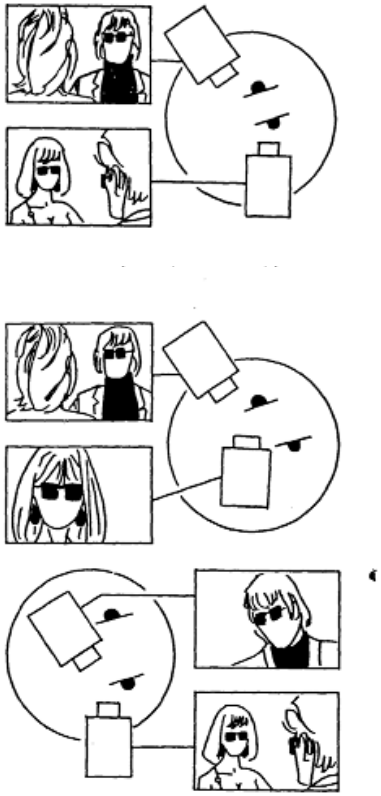
Film

Sequences

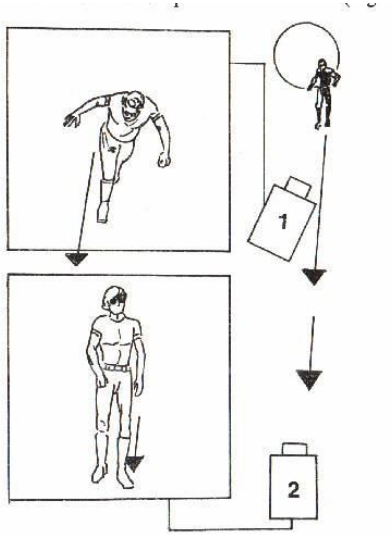
Scenes

Shots

Frames



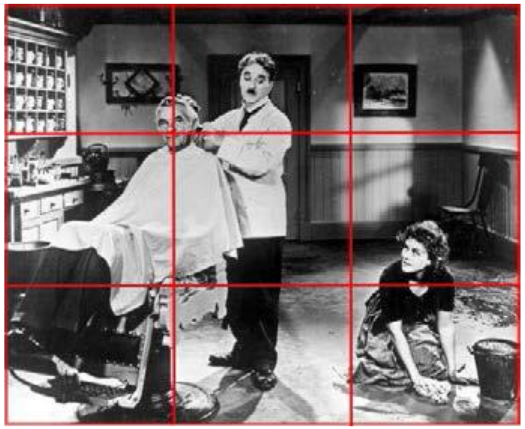
Shot « idioms »
for two people talking
(Arijon 1976)



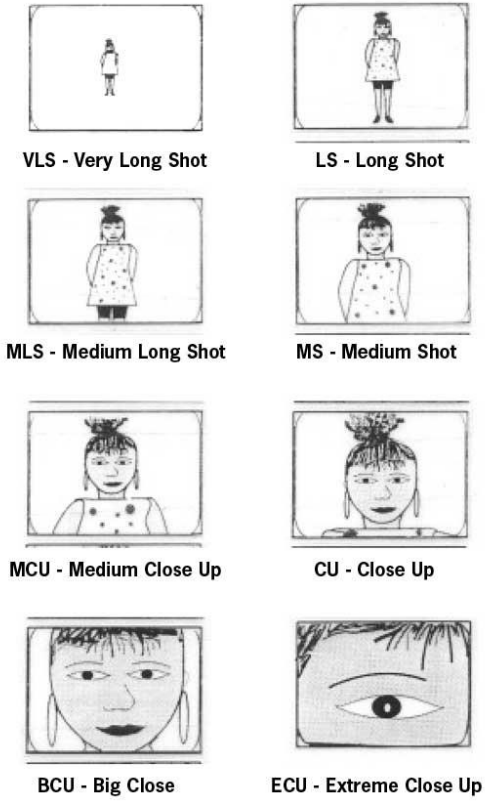
Shot « idiom »
for single character moving
(Arijon 1976)

Transposition to virtual worlds ?

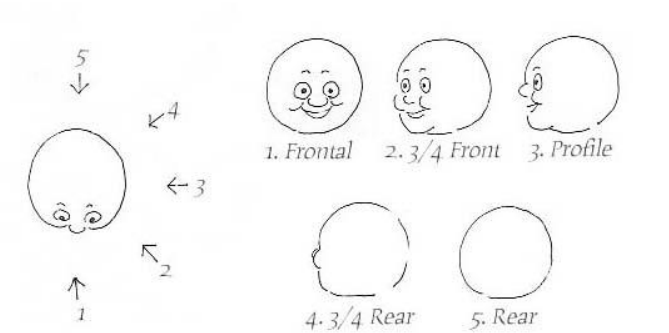
- Key aspect #1 : Viewpoint (frame)
  - Visual composition



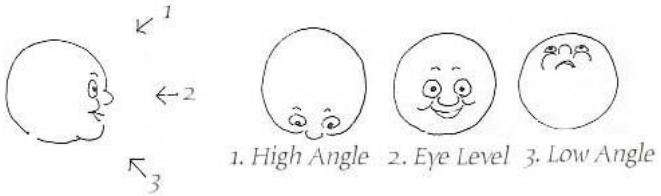
Screen positions



Shot size



Horizontal view angle



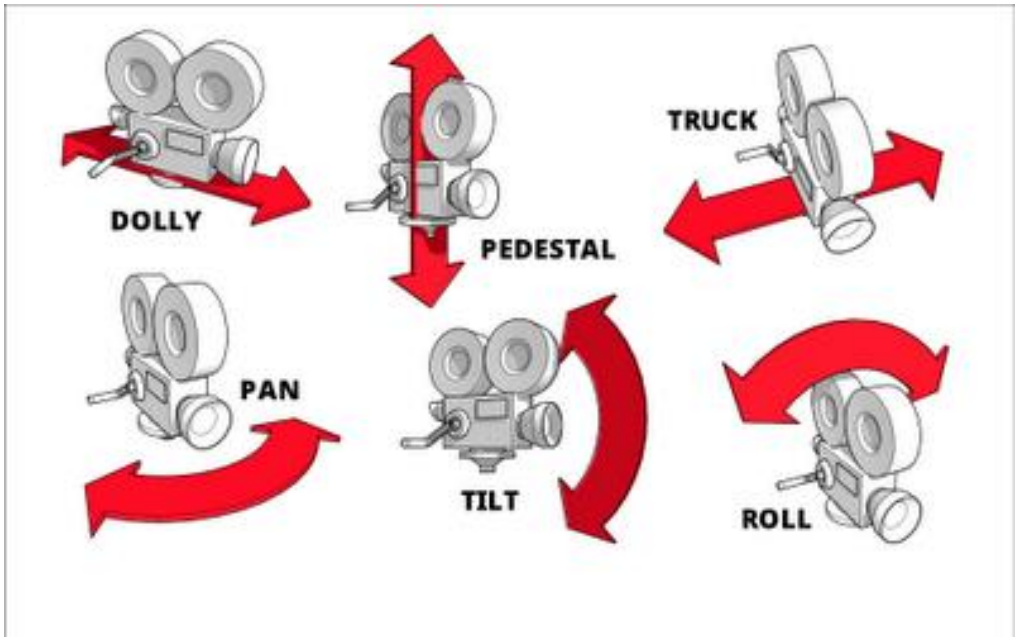
Vertical view angle

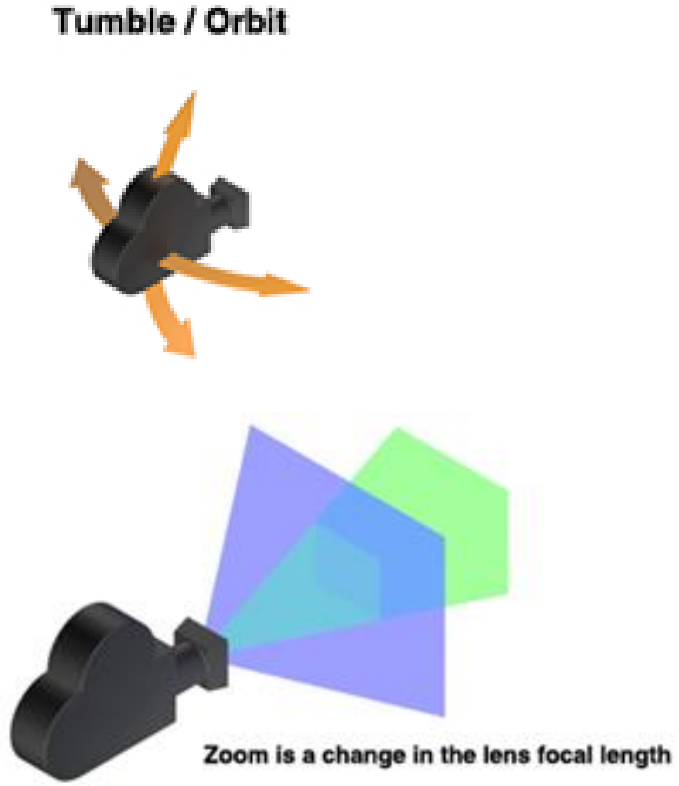
Transposition to virtual worlds ?

- Key aspect #2 : Camera motions

  = (Continuous) sequences of frames / viewpoints

  - Set of key camera motions



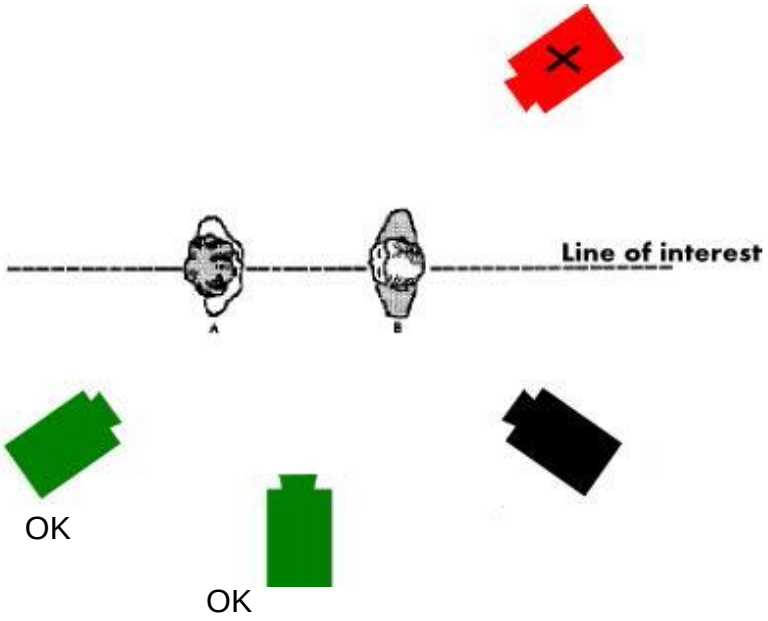


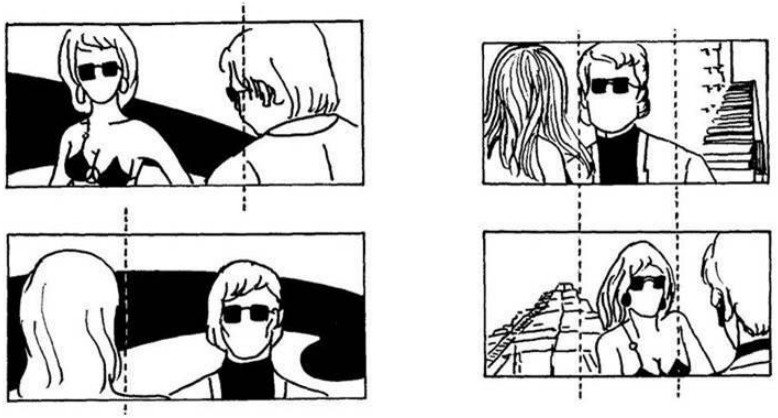
Transposition to virtual worlds ?

- Key aspect #3 : Editing (cut between shots)

  = Sequence of shots

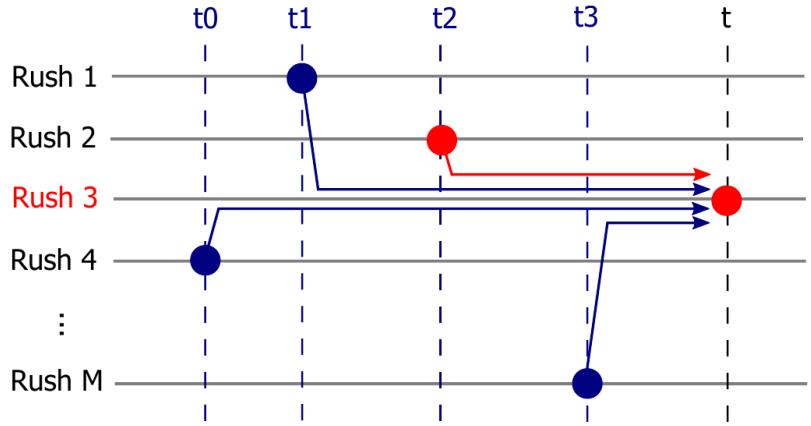
  - Where to cut = Continuity-editing rules
  - When to cut = Overall pace + « cut on actions »

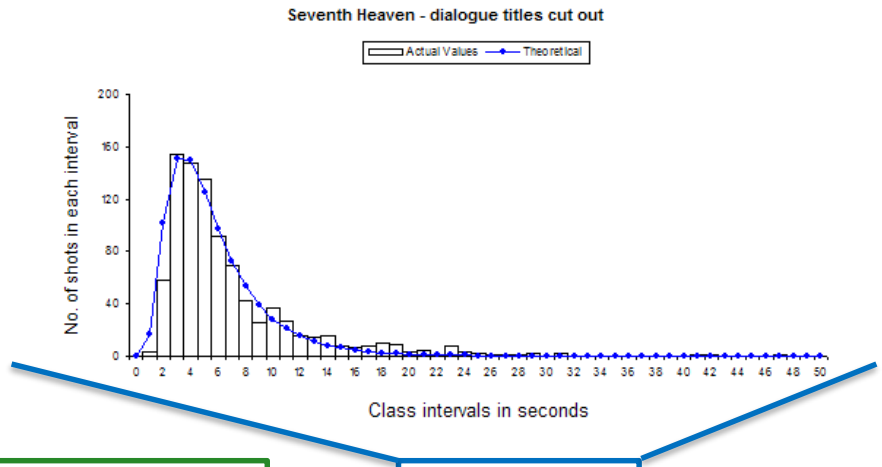




Data-based Editing : « Editing Graph »

- **Editing as a semi-Markov process** (Galvane et al. 2015a)
  - Dataset of rushes: annotated real / virtual footage
  - Cast film grammar as costs : Framing (shots) + Editing (cuts, pace)





$$P(s) = W^A.\sum_j \sum_{t_j \le t \le t_j + d_j} P^A(c_j^t) + W^T.\sum_{1 \le j} P^T(c_{j-1}^t, c_j^t) + W^R.\sum_j P^R(d_j)$$

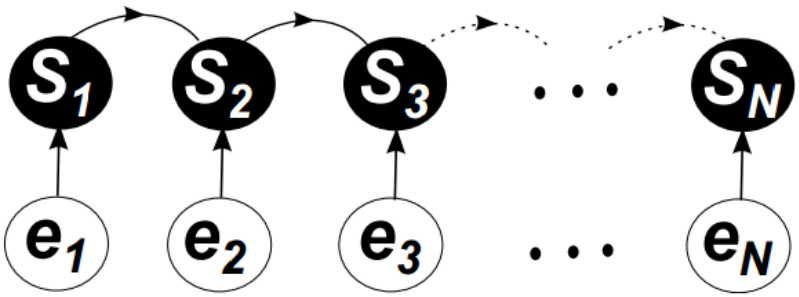
Action cost (Shot quality)

Transition cost (Cut quality)

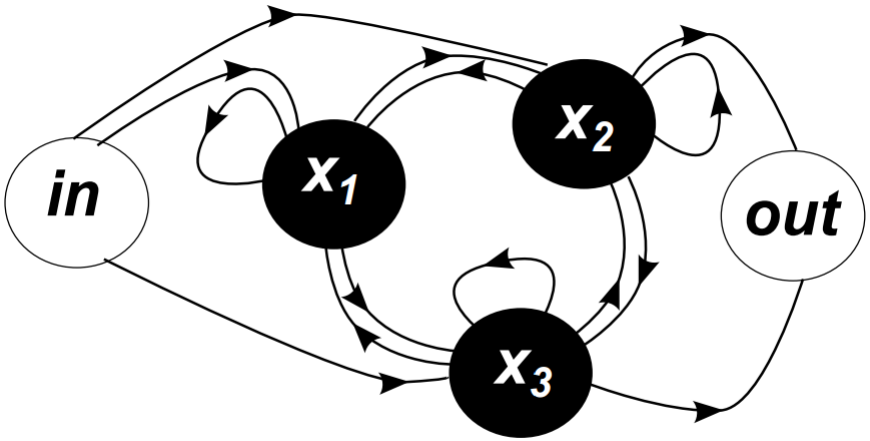
Rhythm cost (Pace quality)

Data-driven Editing using HMMs

- **Learn** elements of style **into an HMM** (Merabti et al. 2016)
  - States = shots, Observations = events
  - Probability of an event : P(event(t) | shot(t))
  - Probability of a cut : P(shot(t) | shot(t-1), event(t))



a) first order dependency shots.



Analyzing Style through Editing Patterns

- **Sequences of >2 shots:** Patterns (Wu & Christie 2016)
  - Language to describe ***Embedded Constrained Patterns*** (ECPs)
  - Pattern recognition = parse shot sequences (string matching)
  - Today: Smart interactive editing tool (relies on such patterns)

```
intensify{
    relation
        constraint: closer
        range: all
    sub-sequence
        constraint: ECP{
            relation
                constraint: same-size
                range: all
        }
        range: all
    length
        constraint: length>=3
}
```

(a) Lord of the Rings: Fellowship of the Ring (2001)

(b) Godfather (1972)

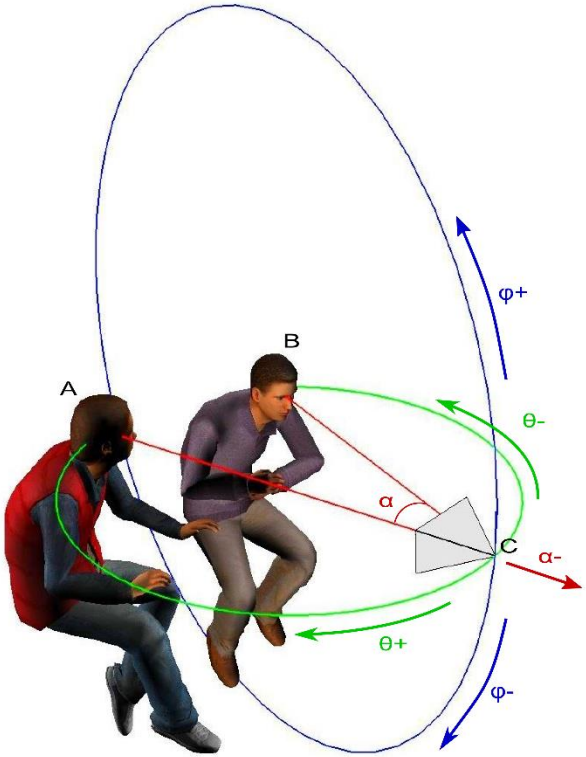
Figure 1: These two sequences in *Lord of the Rings* and *Godfather* both show the intensify ECP (moving gradually closer to the actor).

Toward better using existing film datasets?

- **A priori on edits**
  - Time series: patterns, transition matrices, include camera motions
  - **Not well described just as a time series**
    - Difficult to align two edits and compare
  - Better described as a **set of stylistic features (statistics) ?**
    - Use of key viewpoints
    - Transition matrices (between types of shots + context)
    - Use of a cut rhythm (distribution law)
    - Use of a set of editing patterns (+ frequencies) ?
    - Use of key camera motions ?
  - **Which features** to describe them well: Bi-clustering?

- **Improve dataset of edits** (how big ? today only 22 clips)
  - Manual annotation vs. Automated annotation
  - Group into classes of editing « styles »
  - Extract stylistic elements from the data (Wu et al. 2017)
  - Goal = better understand data → build smarter editing models

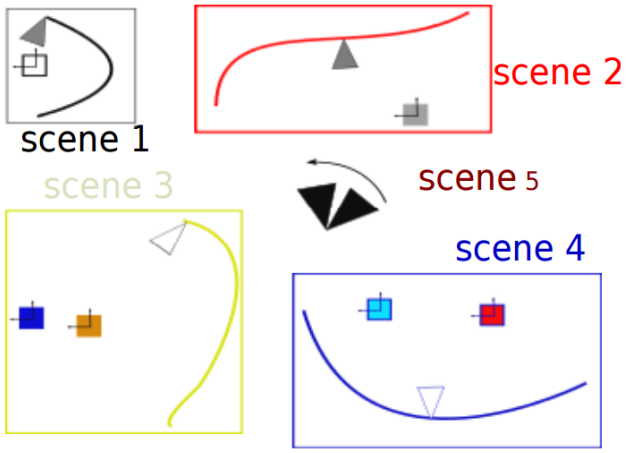
Placing cameras with the Toric Space (Lino & Christie 2015)



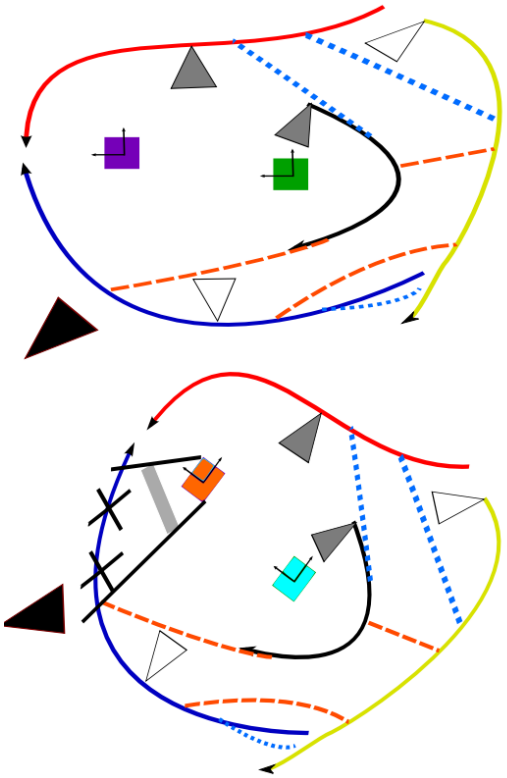
- 7D Camera (world) ➔ 3D camera (local)
- Directly encode some framing properties

Data-driven camera paths

- **Camera motion graph** (Sanokho et al., 2014)
  - Camera paths encoded using the Toric Space
  - Graph of possible camera paths and transitions



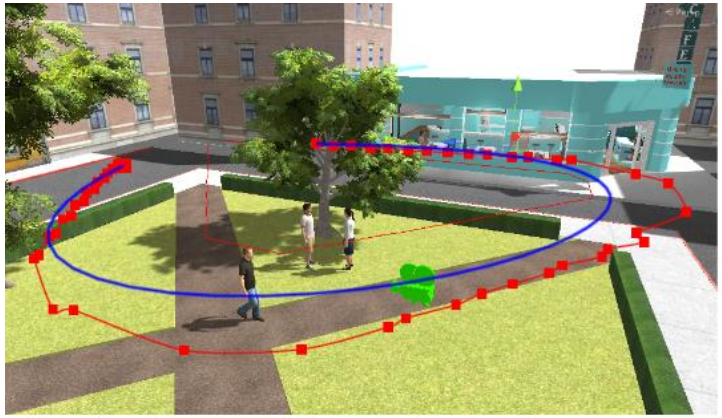
Extraction of camera paths

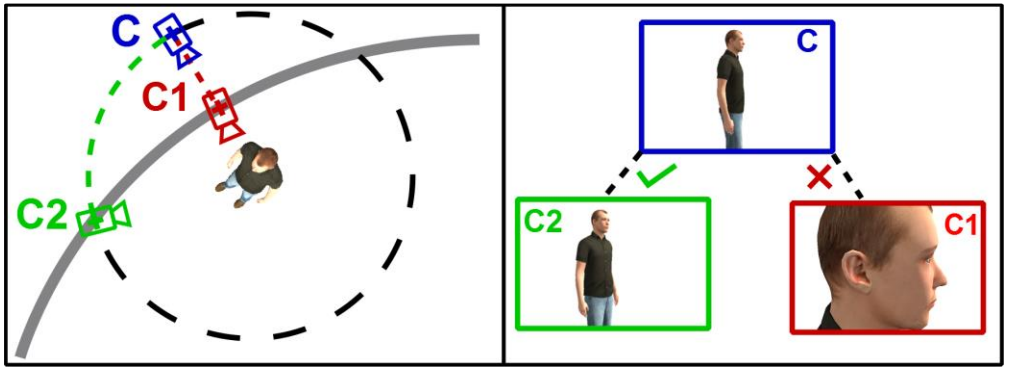


Camera Motion Graph

Data-driven camera paths

- **Constrained camera path: follow characters' motions**
  (Galvane et al., 2015b)
  - Key viewpoints + transitions
  - Computes rails to follow characters' motions
  - Optimize camera motion along the rail





Toward better virtual camera motions ?

- Goal: directly **reuse / adapt existing camera paths** from a dataset
  - Recognize characters' motions or actions
  - Synthtize more realistic / natural camera motions

- **Problems : create classifiers + adapt camera motions**
  - Which feature vector / coordinate system for motions ?
    - **2 complimentary aspects**
      - **Motion in world space** (e.g. is the motion linear or circular ?)
      - **Motion in image space** (change in visual composition ?)
    - Motion relative to filmed characters
    - Motion for N characters ➔ motion for M≠N characters ?
  - **Segment** shots into actors actions/motions + camera motions
  - **Synchronize** / **Link** characters' actions with camera motions
  - **Method(s) for learning / recognizing / retargetting motions ?**

Thank you for your attention

References

[Arijon 1976] D. Arijon: Grammar of the Film Language. Hastings House Publishers, 1976.

[Galvane 2015a] Q. Galvane, R. Ronfard, C. Lino, M. Christie: Continuity-Editing for 3D Animation. AAAI, 2015.

[Galvane 2015b] Q. Galvane, M. Christie, C. Lino, R. Ronfard: Camera-on-rails: Automated Computation of Constrained Camera Paths. ACM Motion in Games, 2015.

[Lino 2015] C. Lino, M. Christie: Intuitive and Efficient Camera Control using the Toric Space. ACM SIGGRAPH, 2015.

[Merabti 2016] B. Merabti, M. Christie, K. Bouatouch: A Virtual Director using Hidden Markov Models. Computer Graphics Forum, 2016.

[Sanokho 2014] C. Sanokho, C. Desoche, B. Merabti, T.Y. Li, M. Christie. Camera Motion Graphs. ACM Symposium on Computer Animation, 2014.

[Wu 2016] H.Y. Wu, M. Christie: Analysing Cinematography with Embedded Constrained Patterns. WICED, 2016.

[Wu 2017] H.Y. Wu, Q. Galvane, C. Lino, M. Christie: Analyzing Elements of Style in Annotated Film Clips. WICED, 2017.