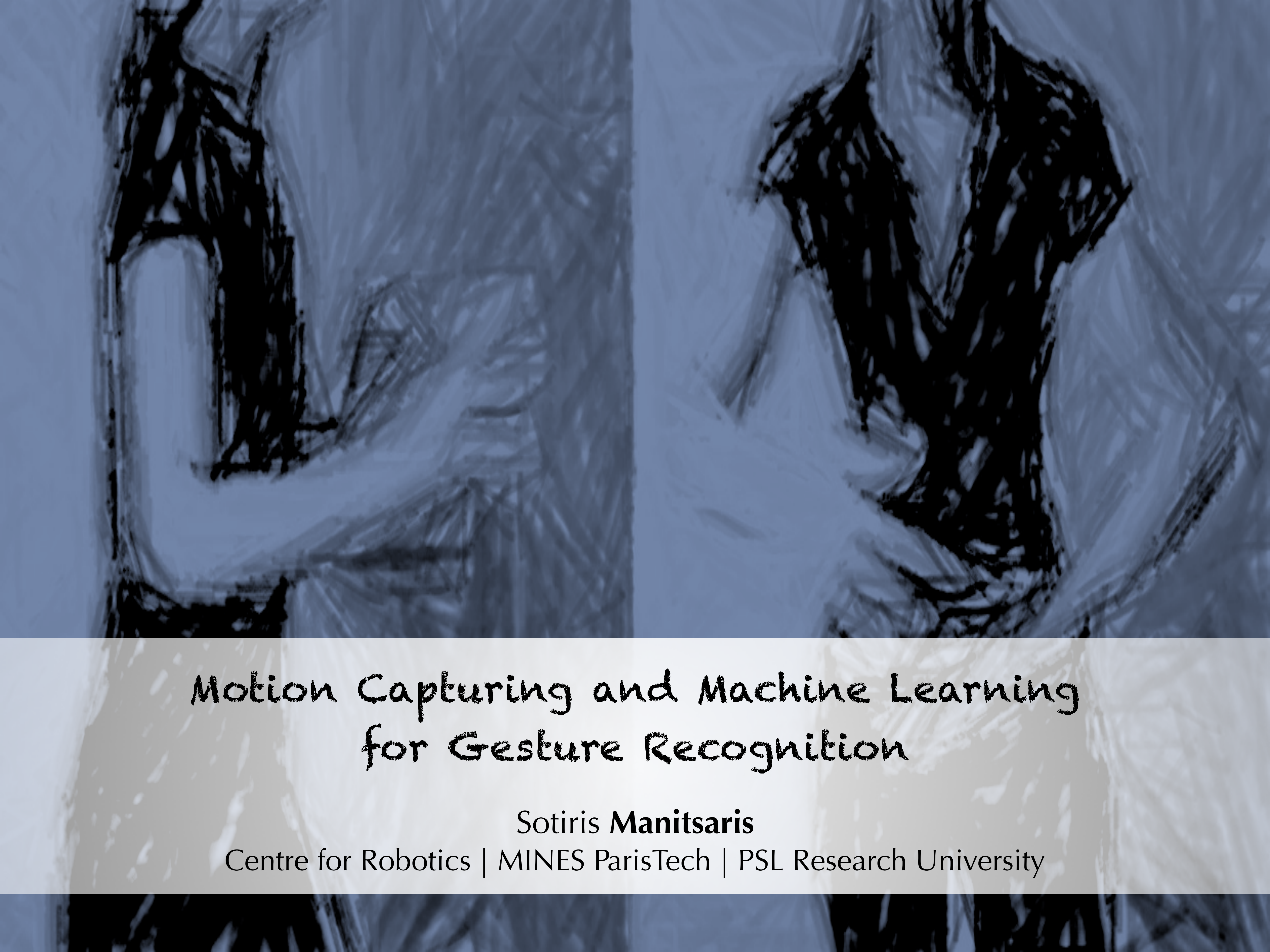
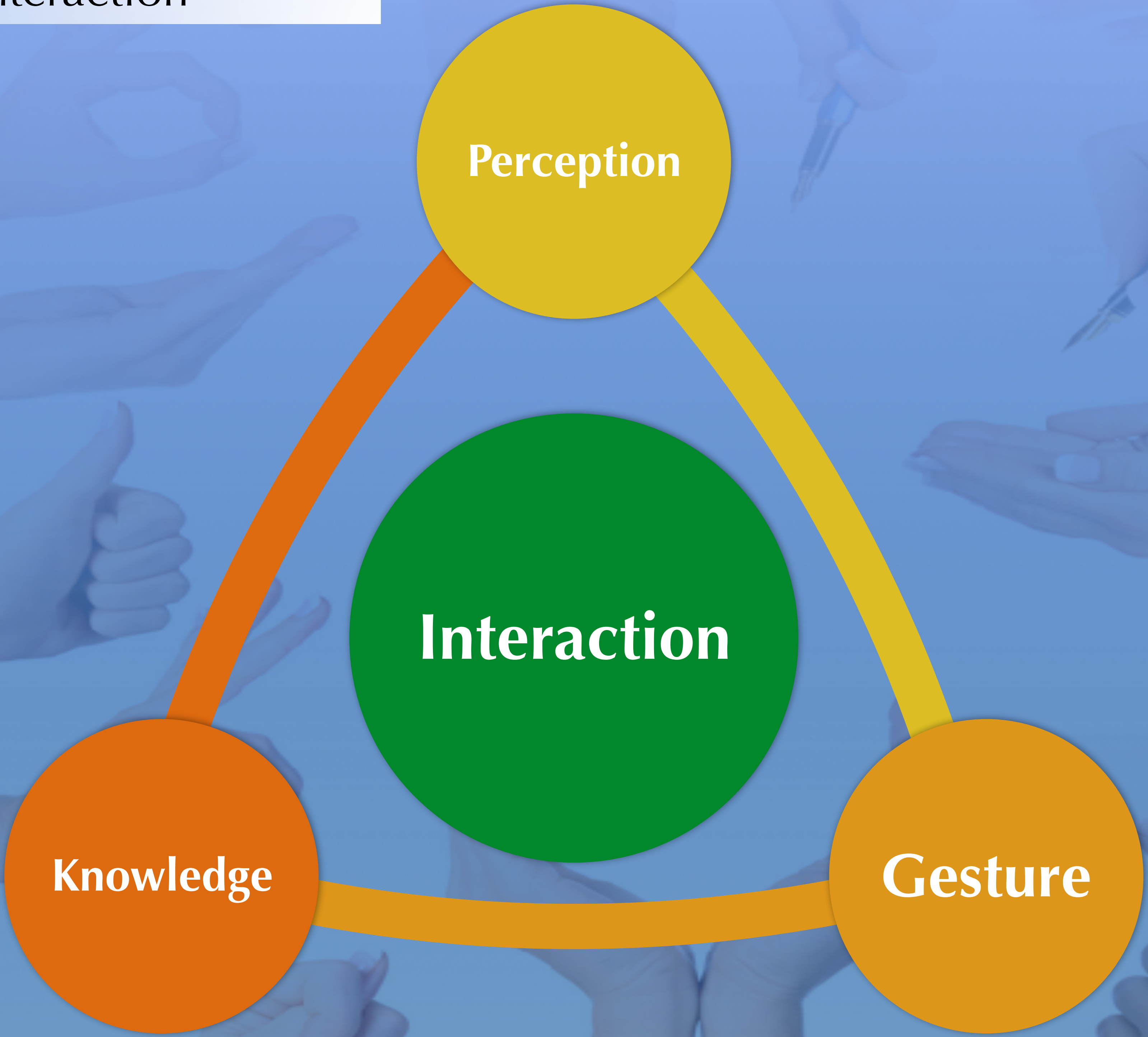


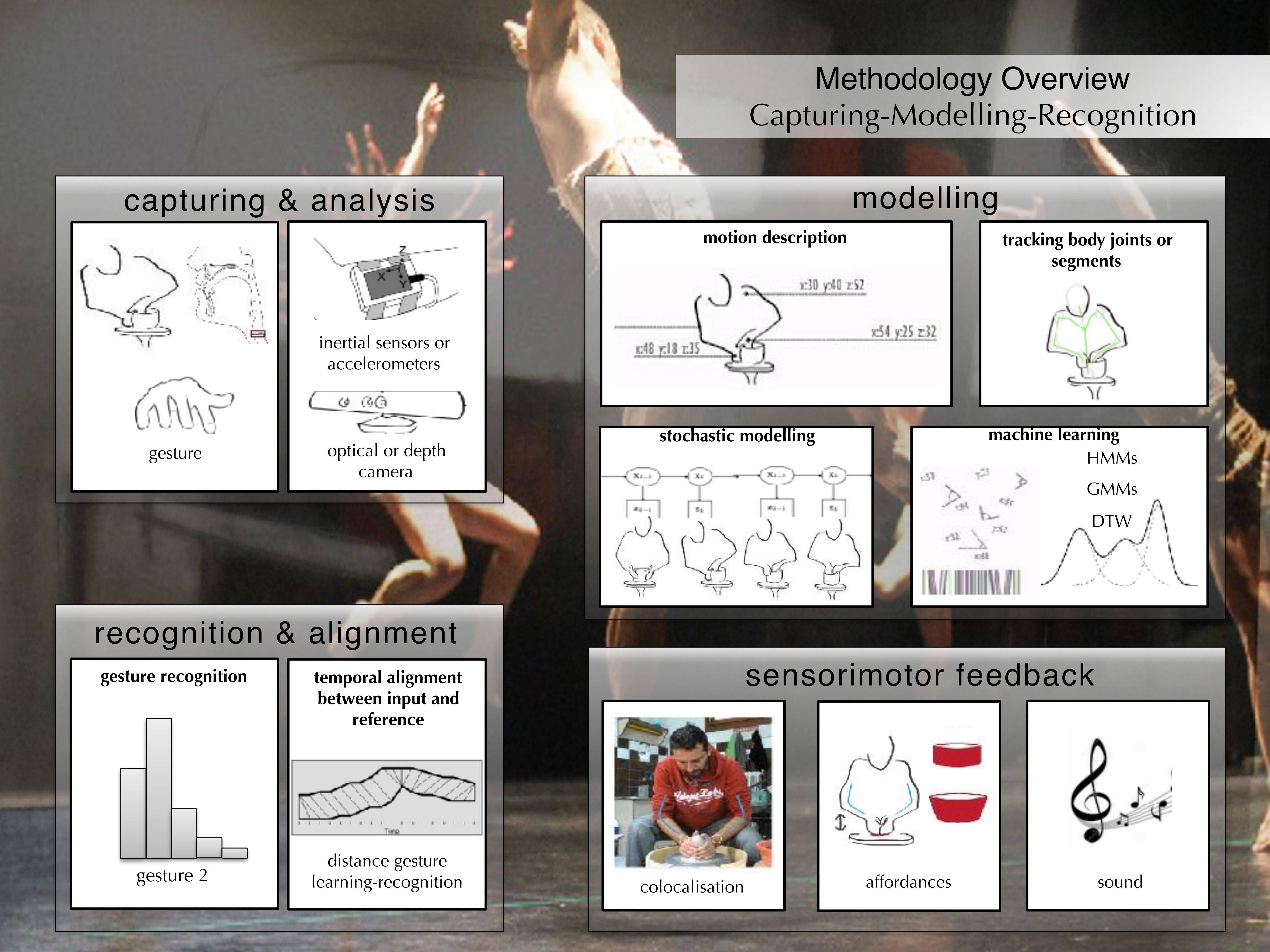
Motion Capturing and Machine Learning for Gesture Recognition

Sotiris **Manitsaris**
Centre for Robotics | MINES ParisTech | PSL Research University

Interactive Systems

Gestural interaction

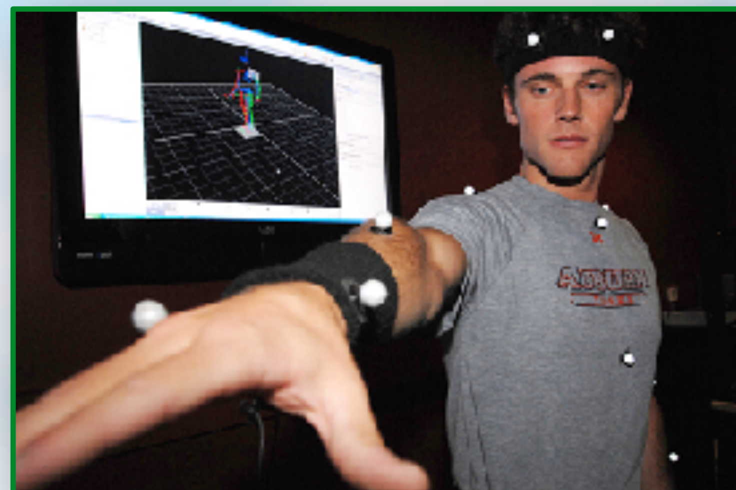


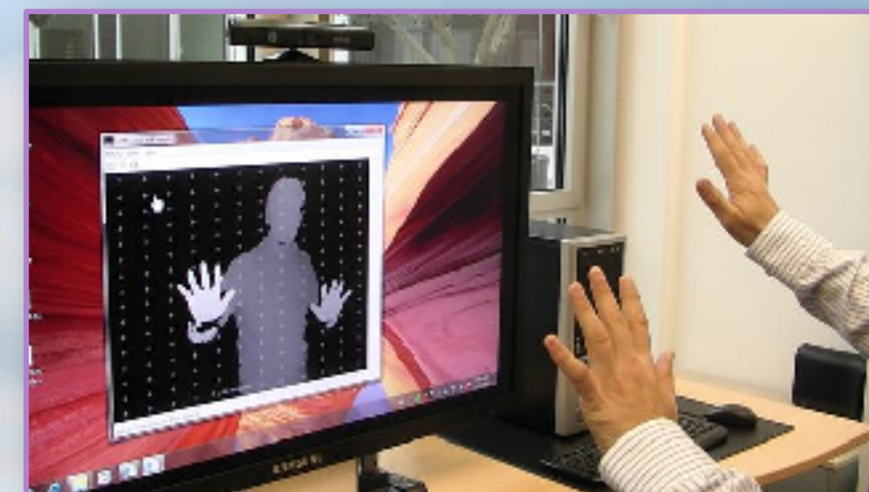


Methodology Overview
Capturing-Modelling-Recognition
capturing & analysis
gesture
inertial sensors or accelerometers
optical or depth camera
modelling
motion description
x:30 y:40 z:52
x:54 y:25 z:32
x:48 y:18 z:35
tracking body joints or segments
stochastic modelling
machine learning
HMMs
GMMs
DTW
recognition & alignment
gesture recognition
gesture 2
temporal alignment between input and reference
distance gesture learning-recognition
sensorimotor feedback
colocalisation
affordances
sound

Motion Capture

Motion Capture
Computer Vision – Sensors





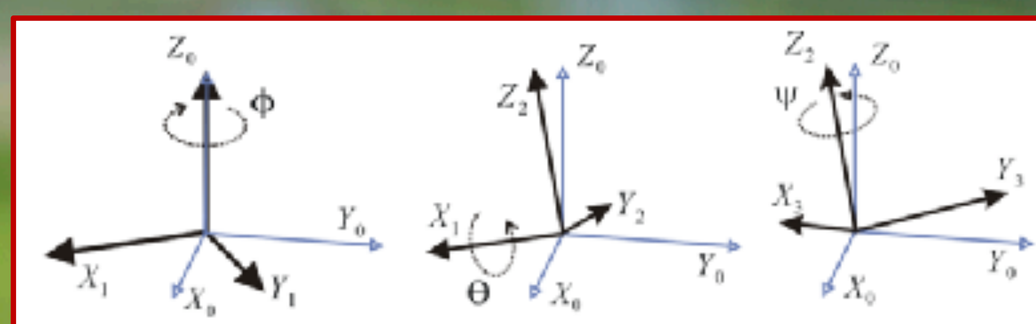
Motion Capture
Wearable or embedded sensors

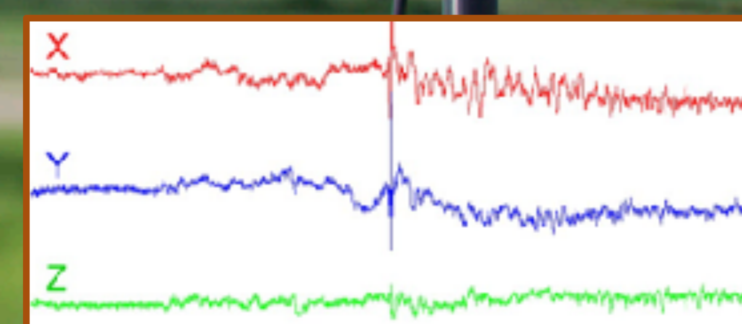
Sensors

- Inertial sensors
  - Magnetometers
  - Gyroscopes
- Accelerometers
- Electromyographs (EMG)

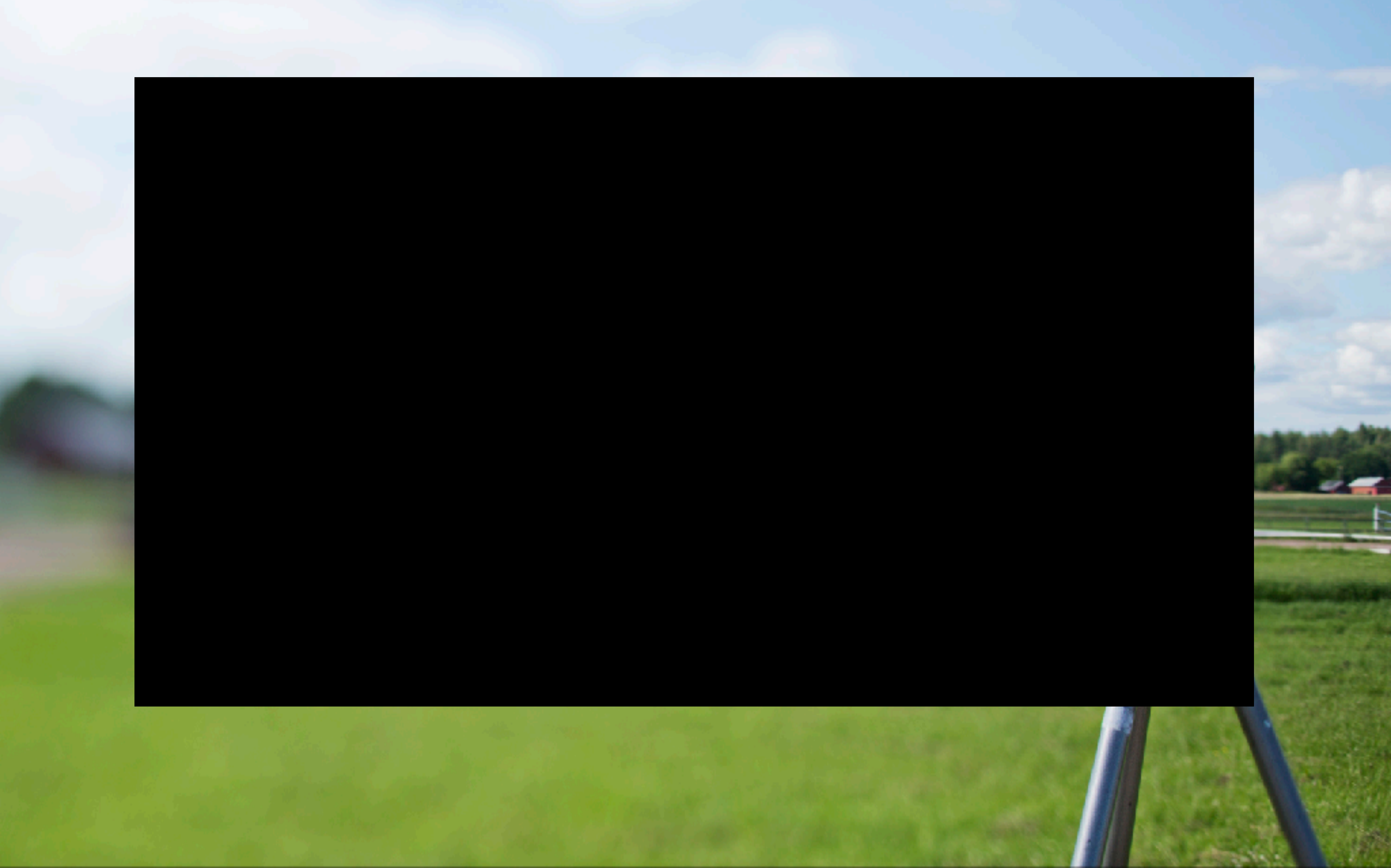
Gestural descriptors

- Rotations
  - Euler angles
  - Axis/Angle
  - Quaternions
  - Exponential map
  - Rotation matrices
- Accelerations

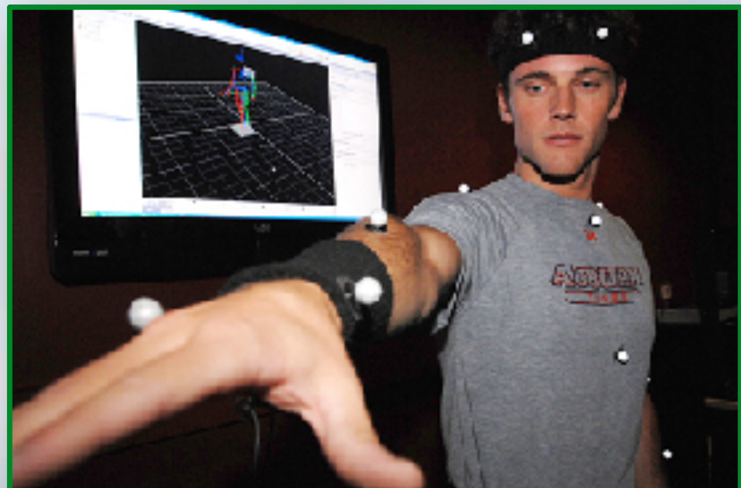


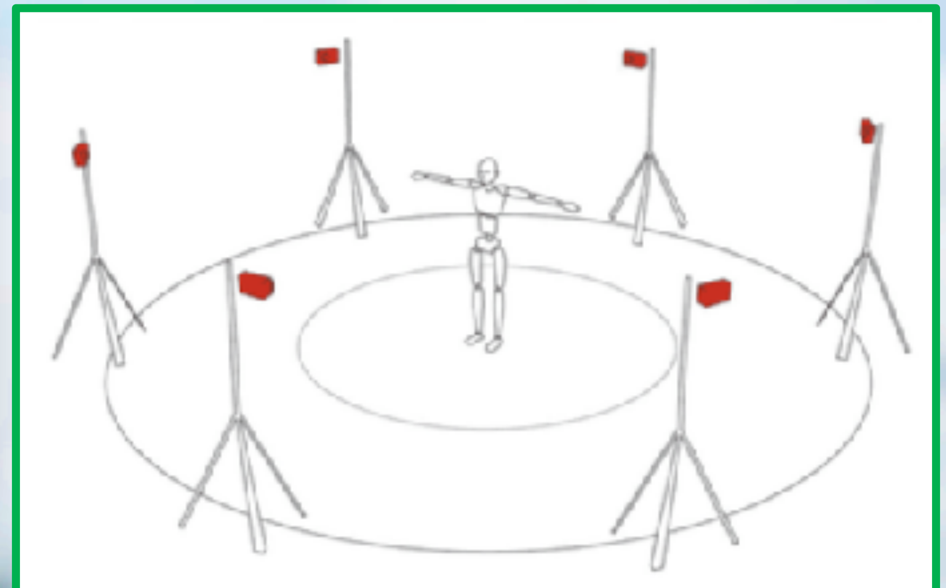


Motion Capture
Wearable or embedded sensors



Motion Capture
Wearable or embedded sensors

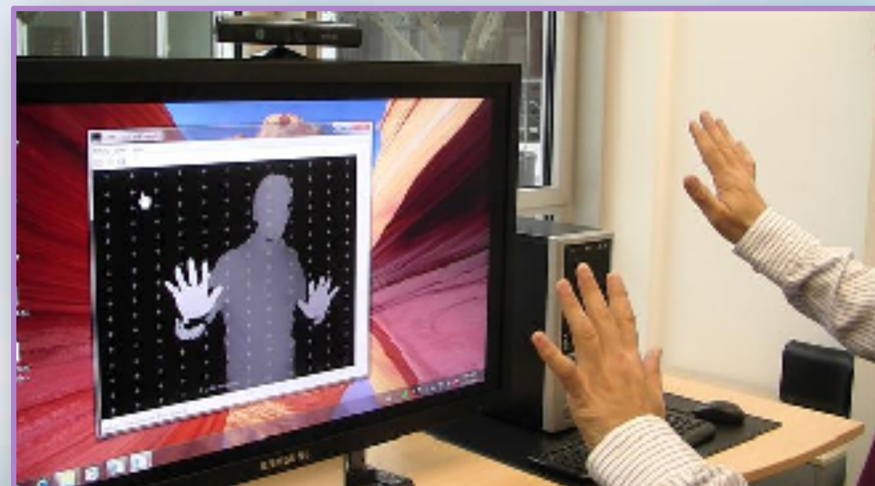


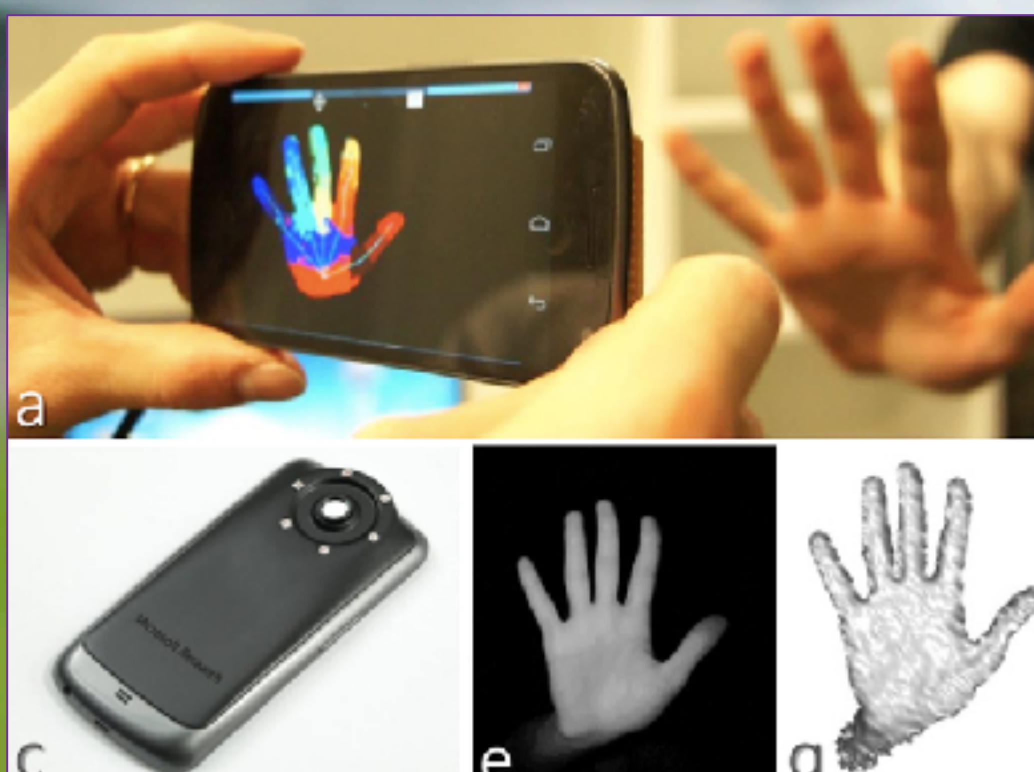


Sensors
- Retroreflective markers
- Light emitting diodes
- Overlapping projections

Gestural descriptors
- Cartesian coordinates

Motion Capture
Markerless computer vision





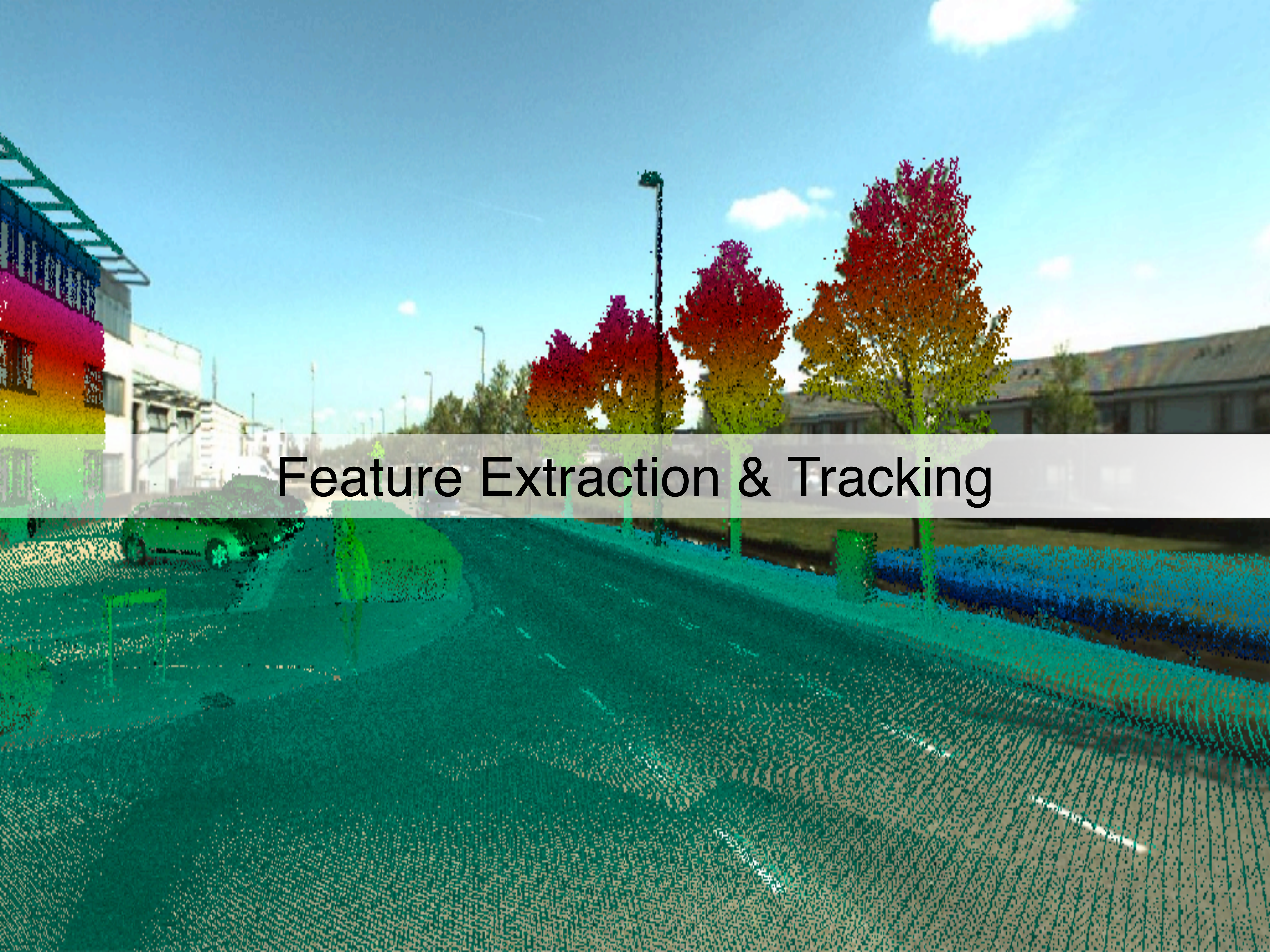
Sensors

- RGB cameras
- Depths cameras

Gestural descriptors

- Cartesian coordinates

Feature Extraction & Tracking



Finger Tracking with RGB Cameras (Musical Interaction)
Skin model and mathematic morphology

Skin modeling

BIP

Échantillonnage

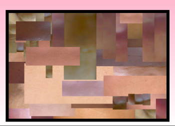
Obtention d'échantillons de couleur de peau et d'ongles P_i

$$P_i(p_j) = \left[R_j, G_j, B_j\right]^T$$

Détermination de la RI

Création d'une image à partir des échantillons P_i

$$RI^{RGB}[m,n]$$



Normalisation de la RI

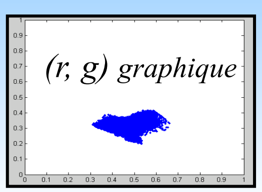
$$\left.\begin{aligned}&\forall p_j^{RI} \in RI^{RGB}\\&N : RI^{RGB} \rightarrow RI^{rg},\\&N([R_j,G_j,B_j]^T) = [r_j,g_j]^T\end{aligned}\right\} \Rightarrow RI^{rg}[m,n]$$

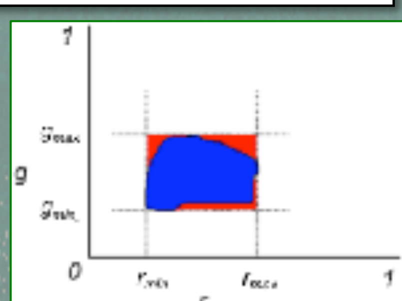
Modèle de la peau

$$\forall p_j^{RI} \in RI^{rg}$$

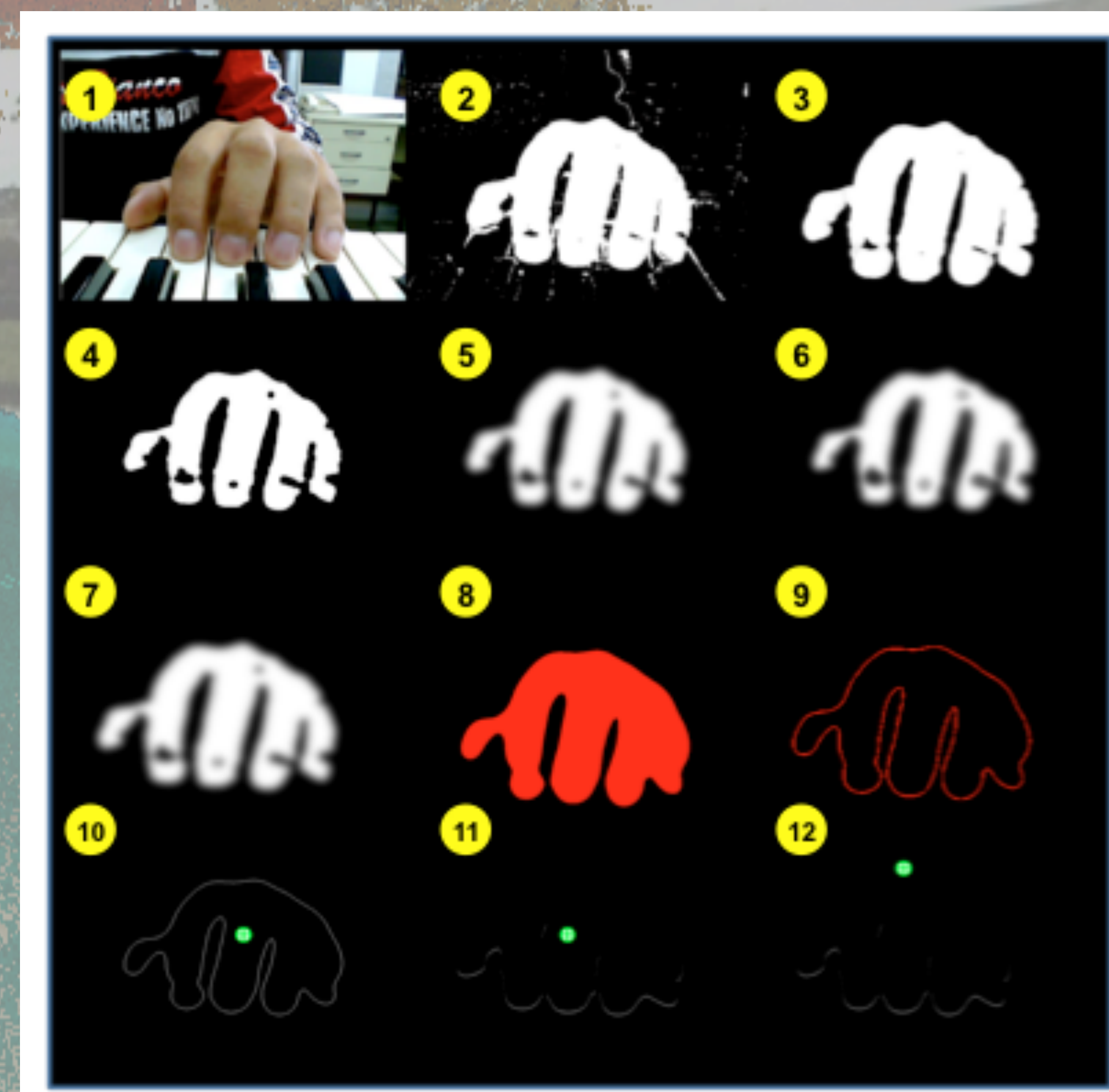
$$r_{peau} = [r_{min}, r_{max}],$$

$$g_{peau} = [g_{min}, g_{max}]$$

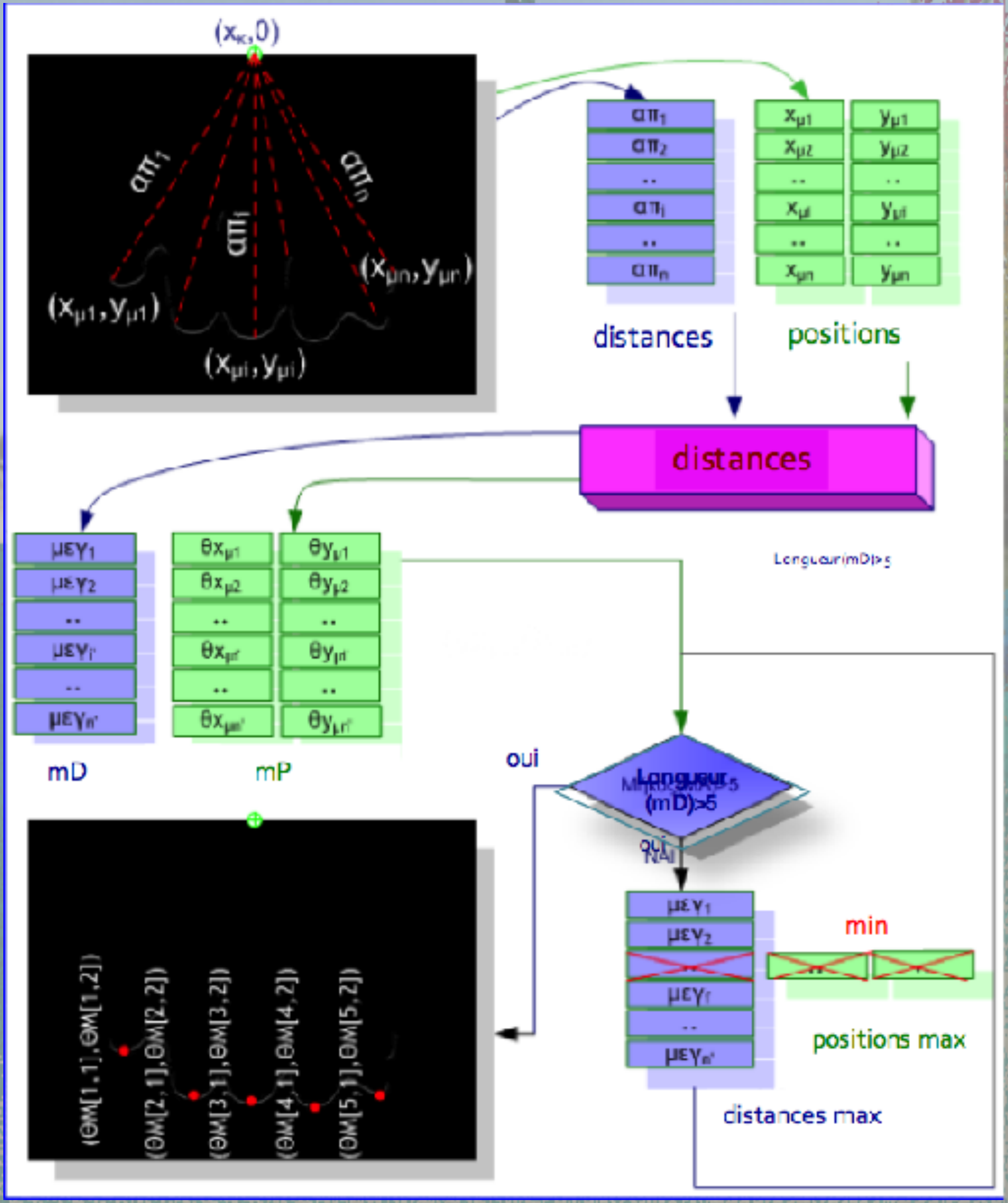




Mathematic morphology and contour detection



Finger Tracking with RGB Cameras (Musical Interaction)
Fingertip Detection



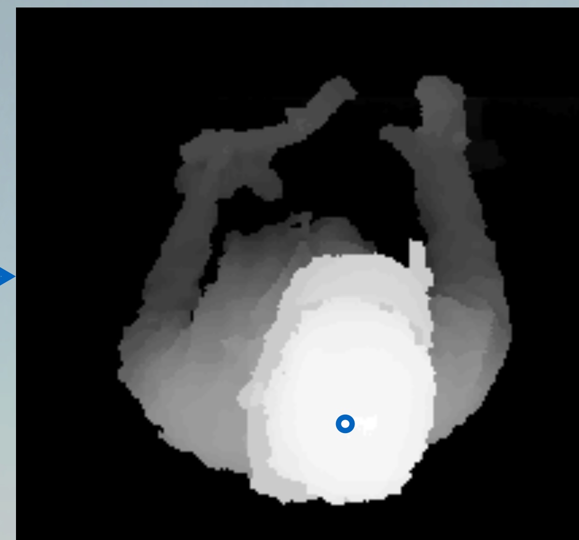
Finger Tracking with RGB Cameras (Musical Interaction)

Real-time finger tracking

Body Tracking with Depth Cameras (Human-Robot Collaboration)

Geodesic distances

Seuillages pour extraire le torse et la position de la tête



Construction d'un graph 2D connectant les pixels du torse

Le poids de chaque arrête est égal à la différence de profondeur entre les deux pixels

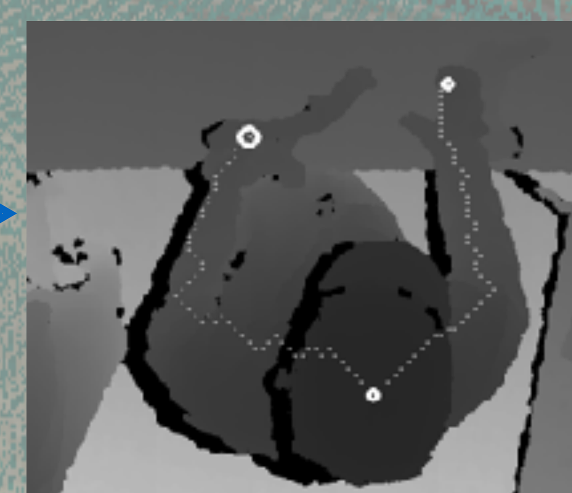
Pour chaque point du torse on calcule le chemin « le plus court » reliant le pixel à la tête

<u>Algorithme de Dijkstra</u> :
Trouver le chemin le plus court i.e. le chemin ayant le poids le plus faible possible
Poids du chemin = Somme des poids des arrêtes parcourues par le chemin

Distance géodésique d'un point du torse à la tête
=
Poids le chemin le plus court reliant ce point à la tête

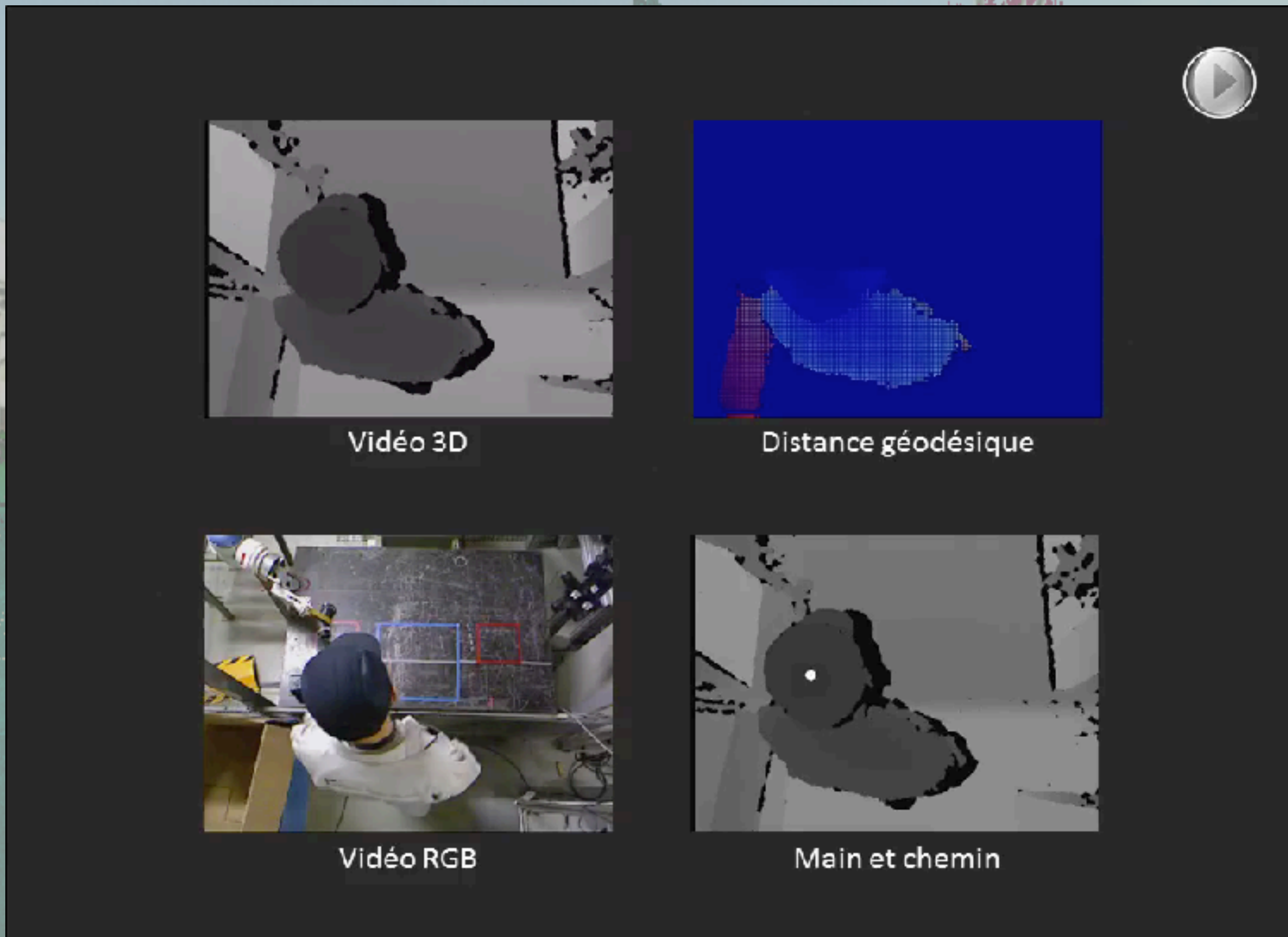
Seuillage pour obtenir les parties les plus éloignées de la tête

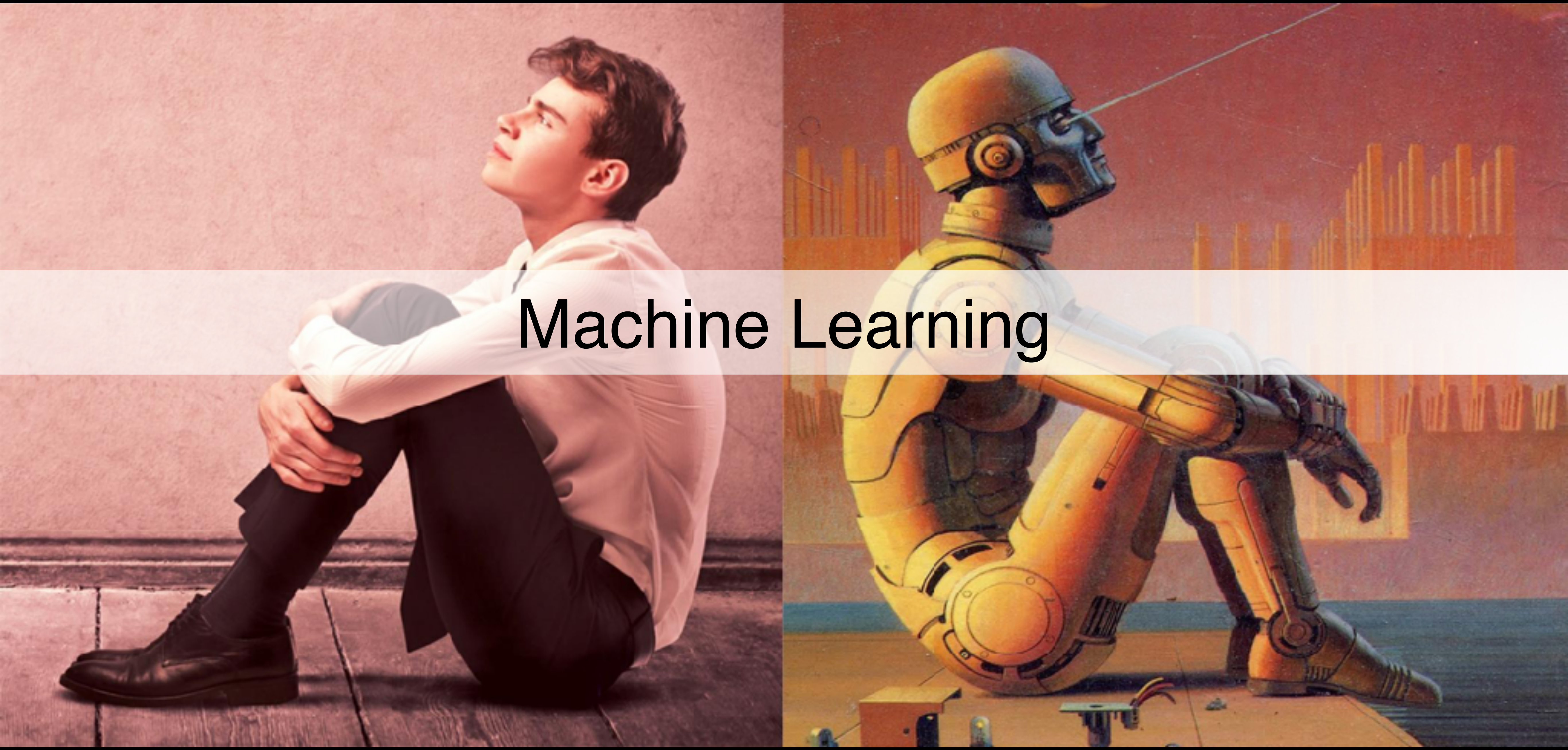
Positions des mains et chemins les plus courts reliant la tête aux mains



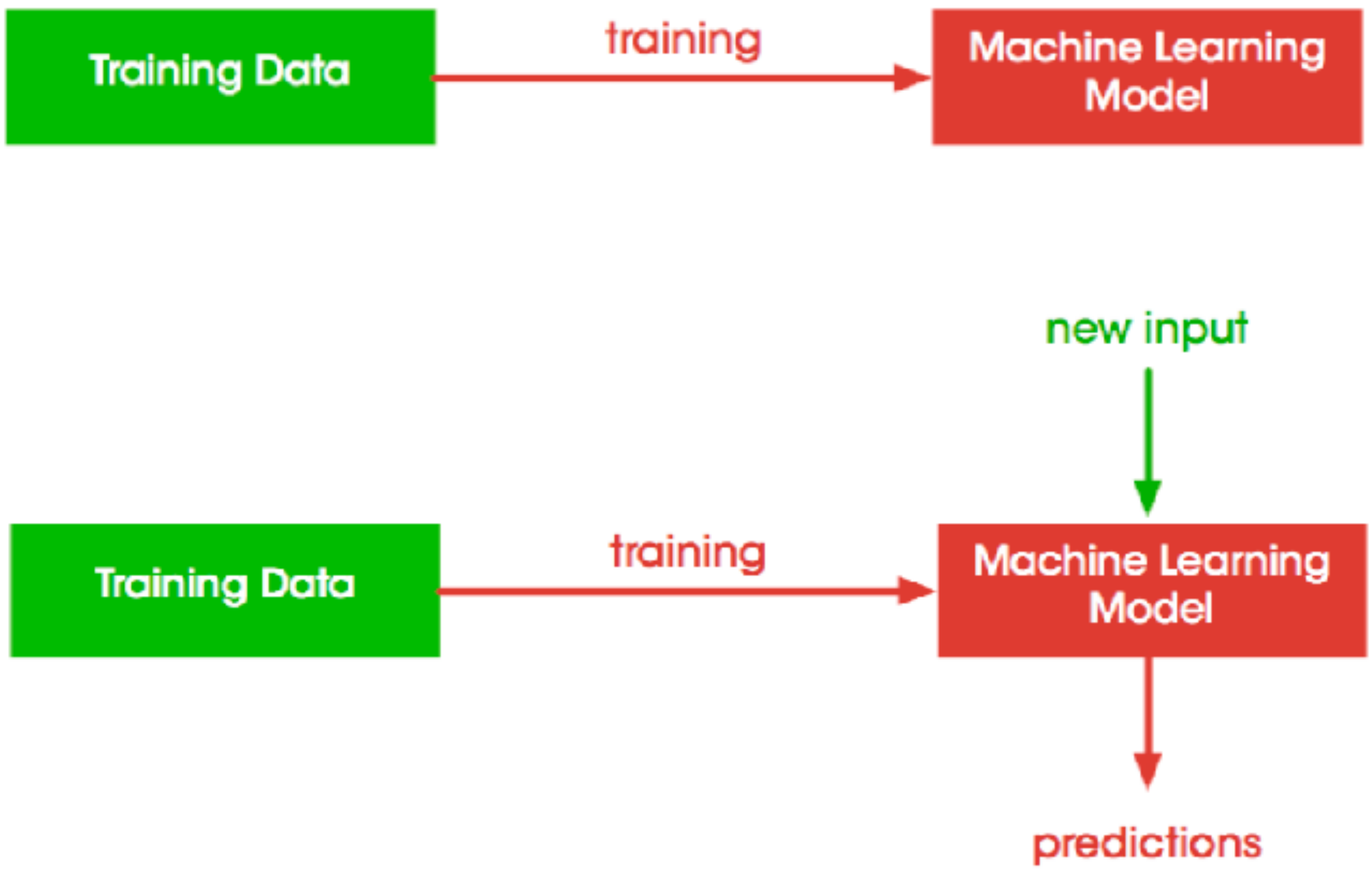
Body Tracking with Depth Cameras (Human-Robot Collaboration)

Real-time body tracking with geodesic distances



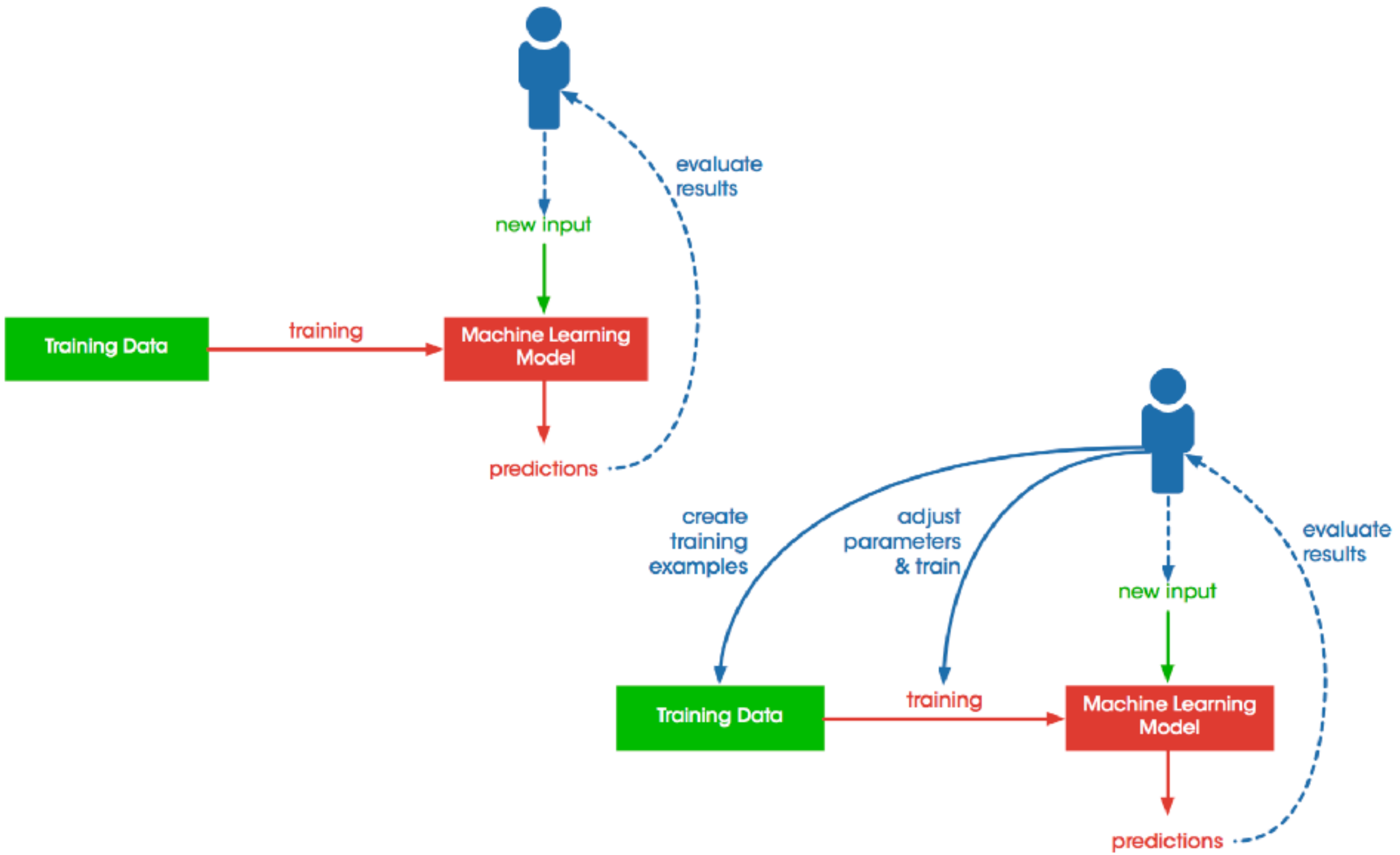

Machine Learning

Machine Learning in Gesture Recognition
Introduction

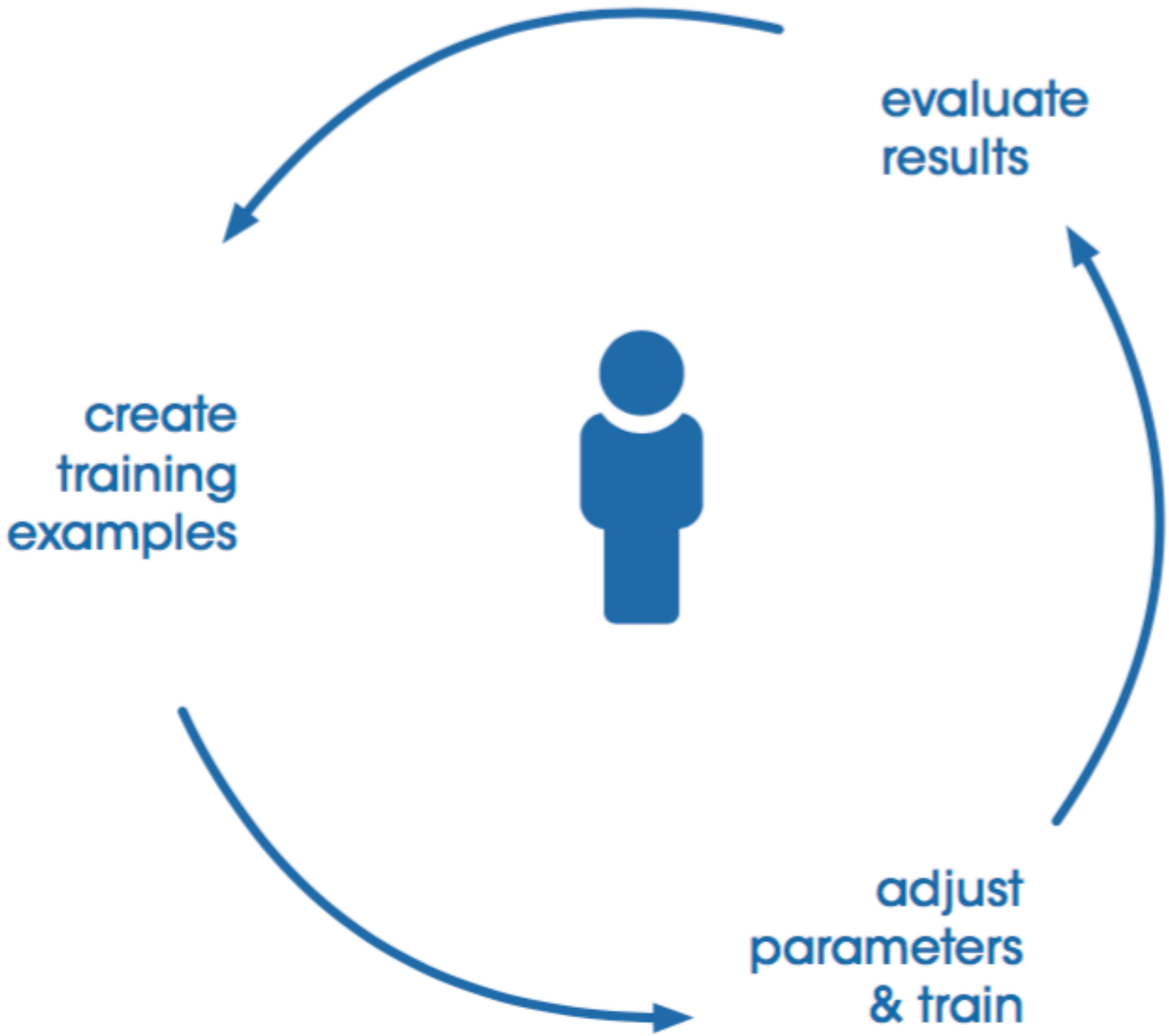


Credits: Jules Françoise

Machine Learning in Gesture Recognition
Introduction



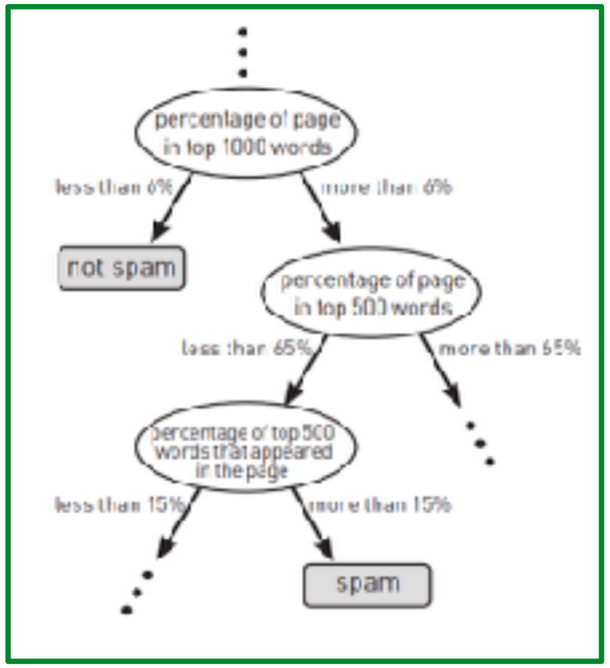
Credits: Jules Françoise

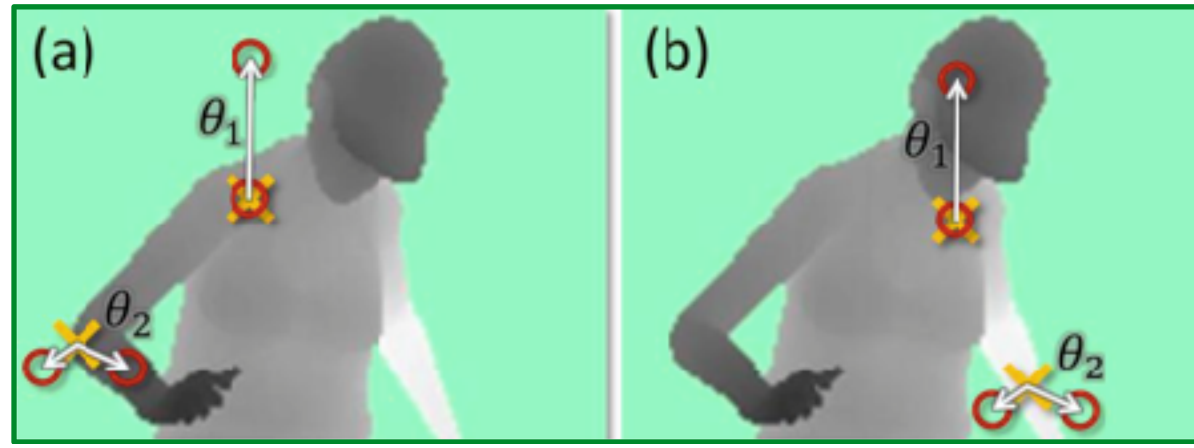


Credits: Jules Françoise

Feature Extraction & Tracking using Machine Learning
Random Decision Forest

Example of pre-planned questions of a decision tree

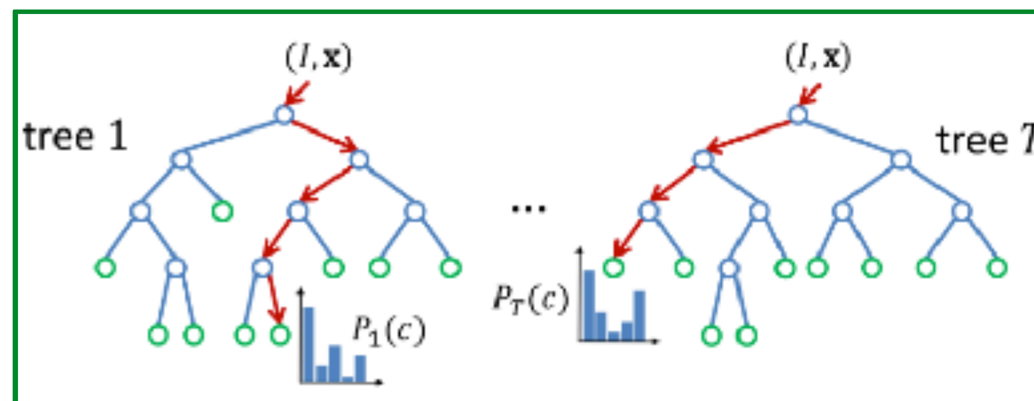




How does the depth at that pixel compare to this pixel?

Random Decision Forest

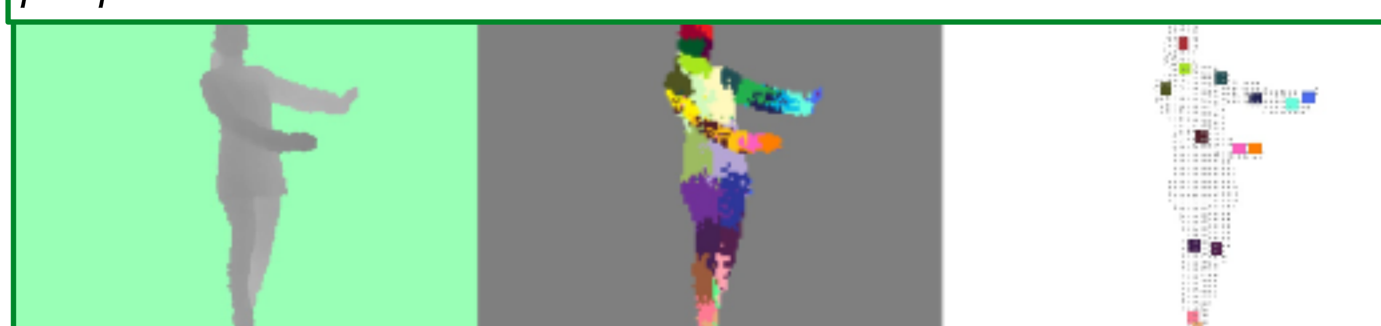
- Use a *random* selection of questions each time
- Learn multiple trees
- Add probability distributions as outputs of the trees to classify



Training the RDF with synthetic images

Tracking the body parts

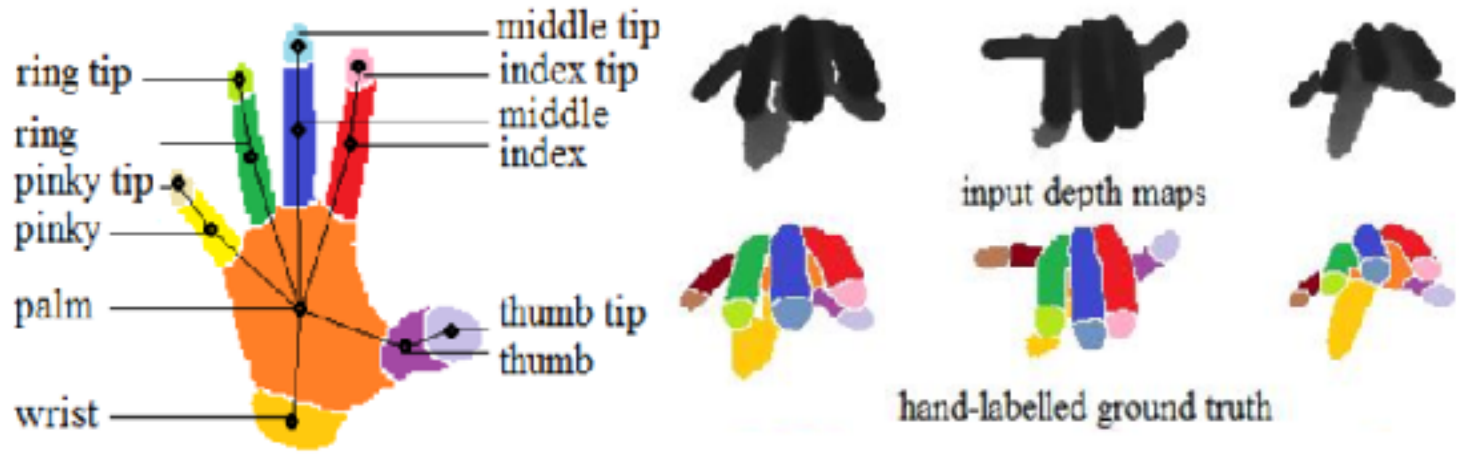
Depth images *Body parts* *3D joint proposals*



Body Tracking with Depth Cameras (Musical Interaction)

Random Decision Forest

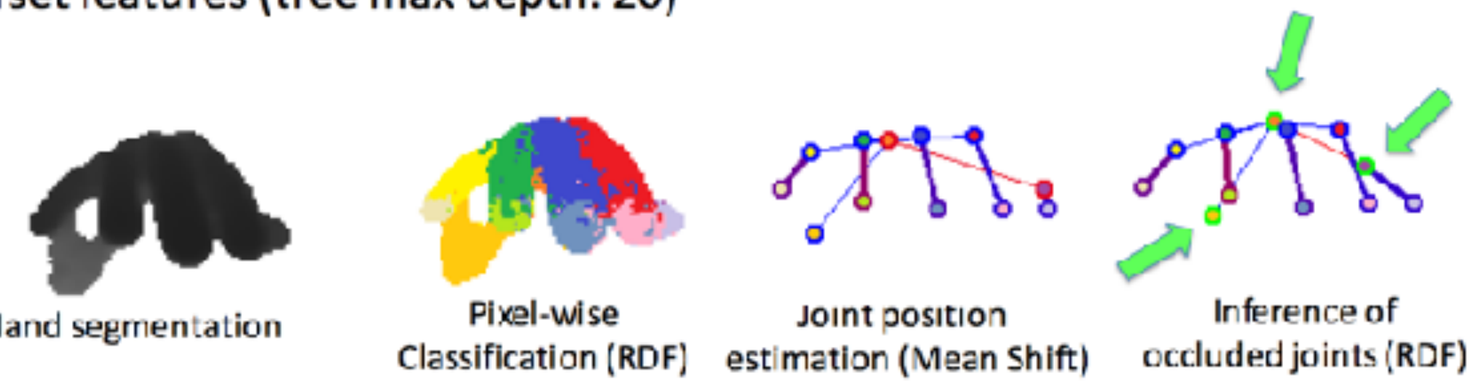
Hand labellisation for classification



- 12 labels [Compared to Shotton 31 labels and Ceskin 19 labels]
- 3 levels model (hand, finger, fingertips) [Compared to the 2 levels of Manitsaris et al.]

Pixel-wise Classification and Joint Estimation

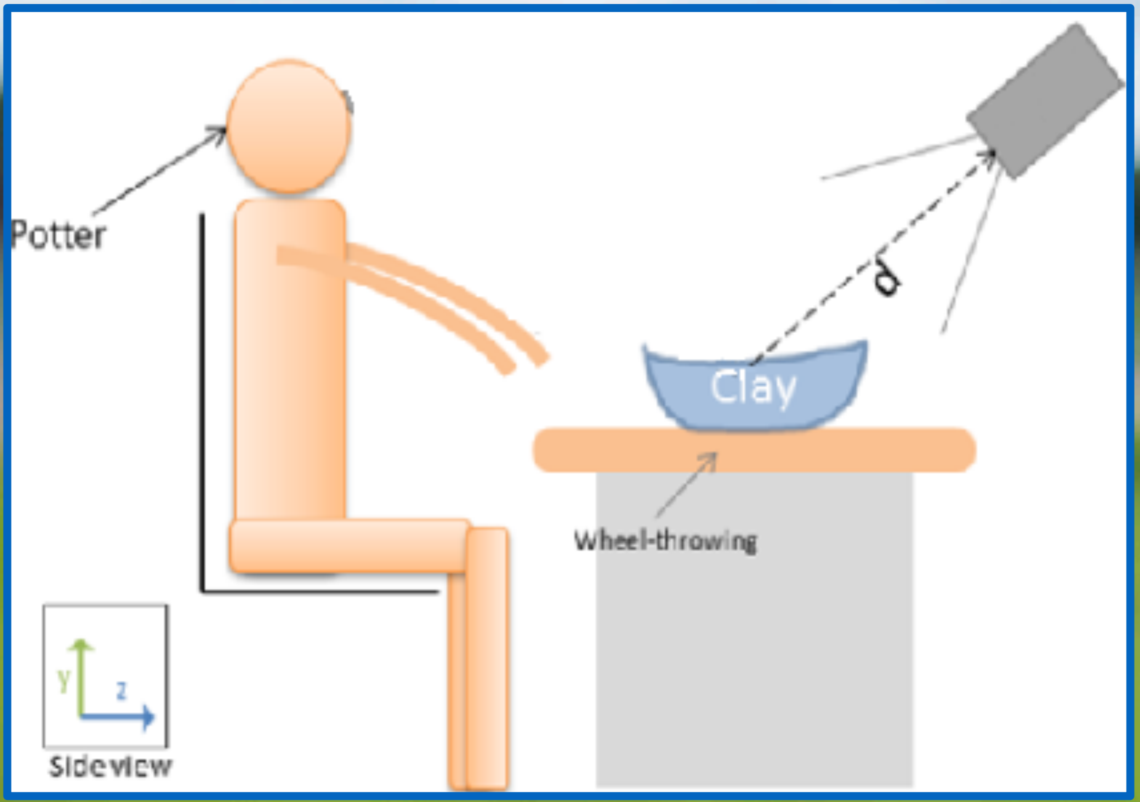
- Pixel-wise classification through Random Decision Forests (RDFs) using normalized depth offset features (tree max depth: 20)



Body Tracking with Depth Cameras (Professional Gestures)
Hierarchical Random Decision Forests

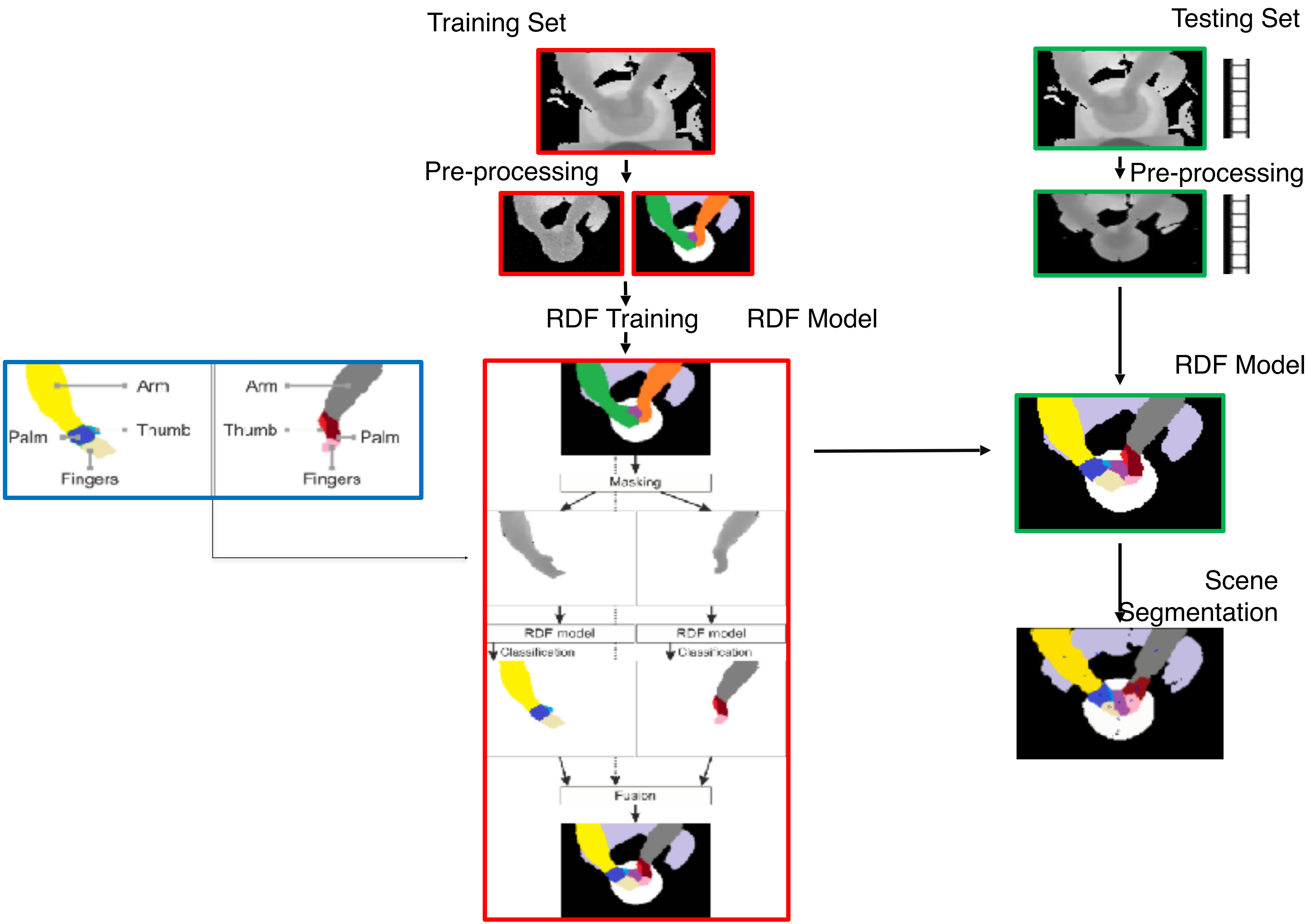
Purpose & Challenges

- Classification of complex scene segments based on machine learning
- The object is Moving, Revolving, Deformable



Body Tracking with Depth Cameras (Professional Gestures)

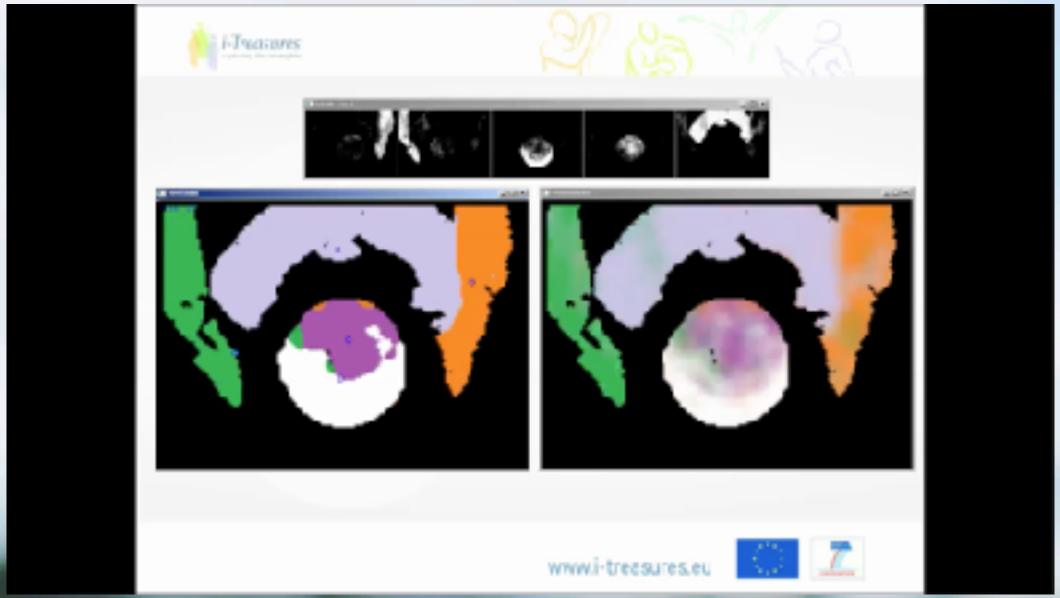
Hierarchical Random Decision Forests



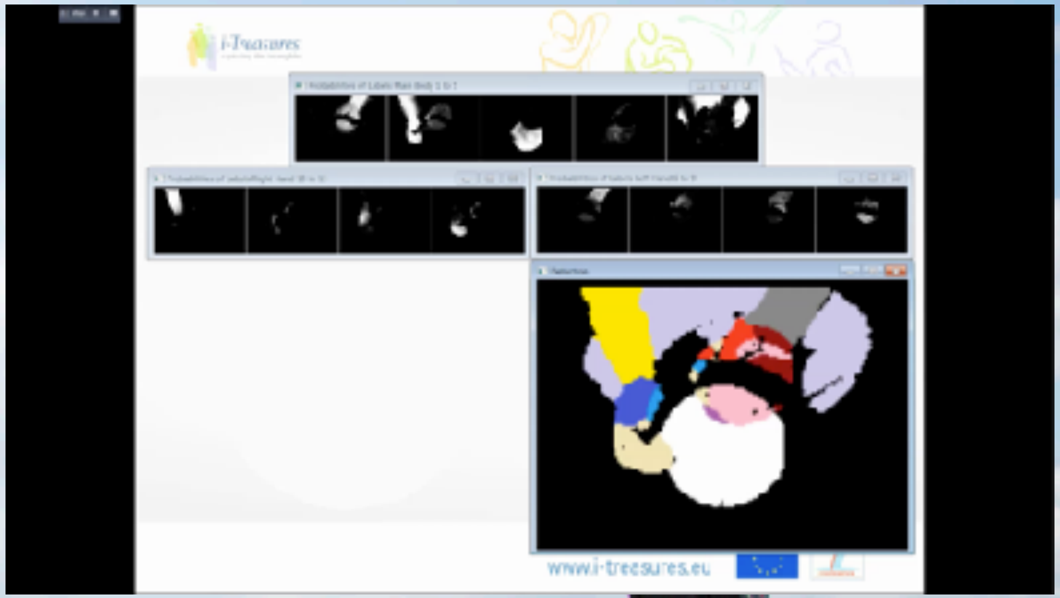
Body Tracking with Depth Cameras (Professional Gestures)

Hierarchical Random Decision Forests

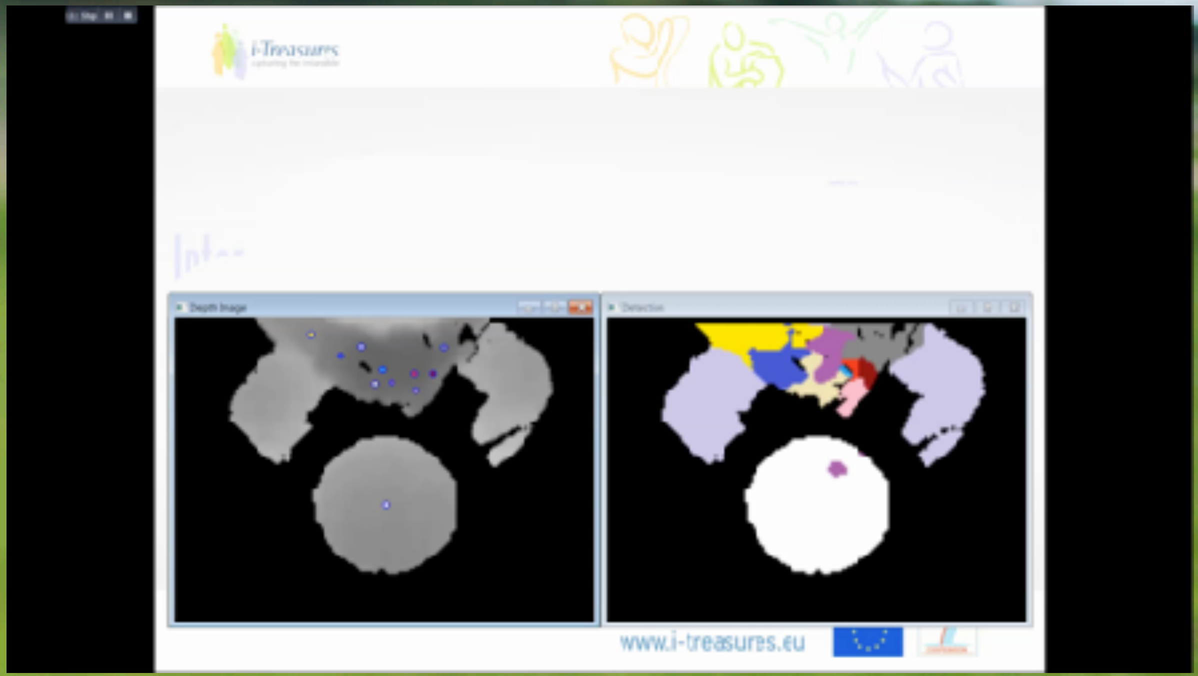
Labels of Parent RDF



Maximum probabilities of labels



Tracking of segments



Full Upper-Body Tracking with Depth Cameras (Intangible Musical Instrument)

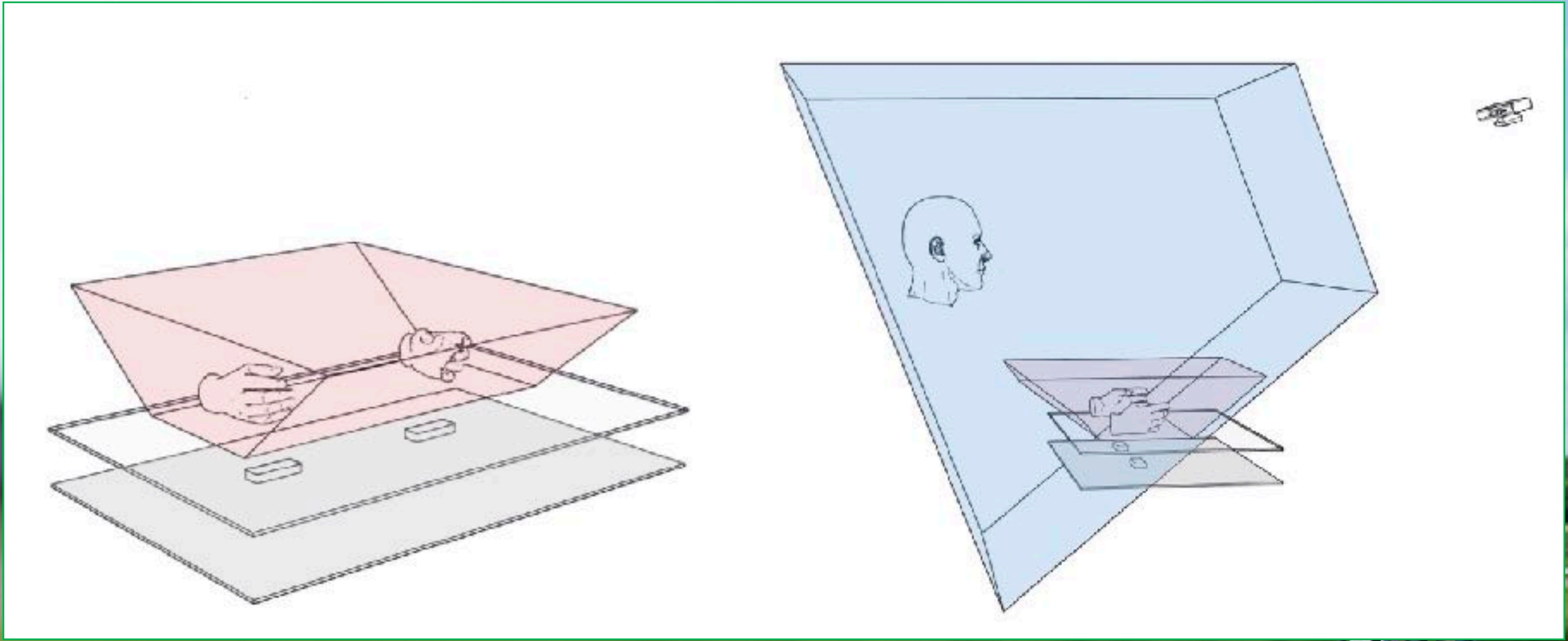
Interactive Space & Surface

Purpose & Challenges

- Natural-User Interfacing the gestural expression and emotion elicitation in music
- Learning, performing and composing with gestures as a first-person experience
- Augmenting the music score to facilitate the access to musical ICH

Full Upper-Body Tracking with Depth Cameras (Intangible Musical Instrument)

Gestures & Embodiment

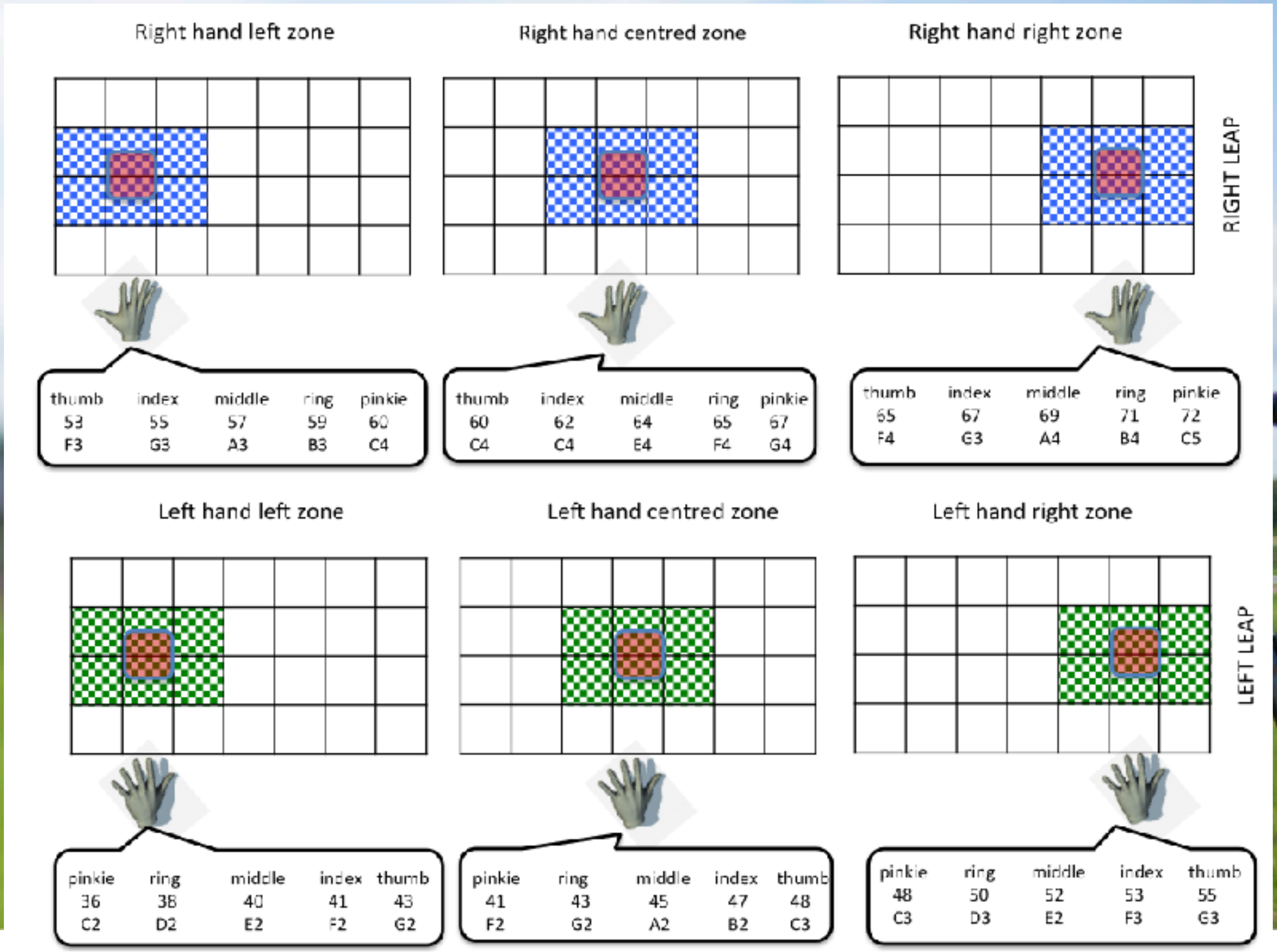


MICRO BB
The Leap motions bounding box (red)
is associated with fingers interaction

MACRO BB
The Kinect bounding box (blue)
is associated with upper-body interaction

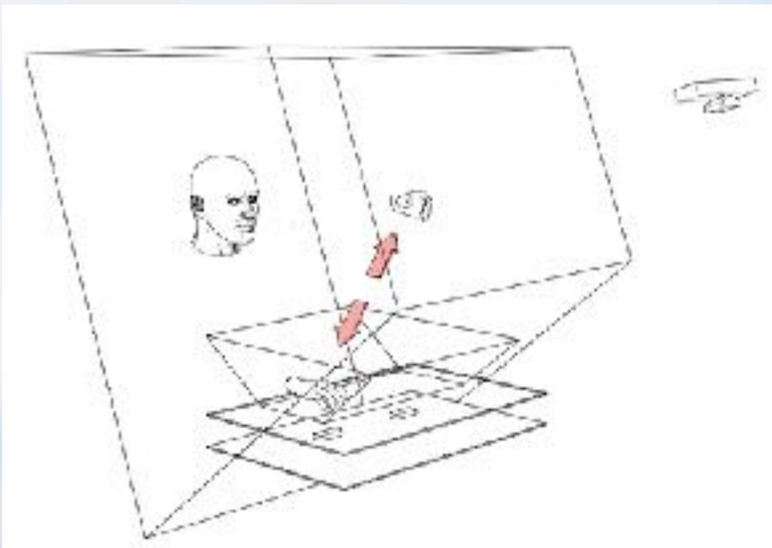
Full Upper-Body Tracking with Depth Cameras (Intangible Musical Instrument)

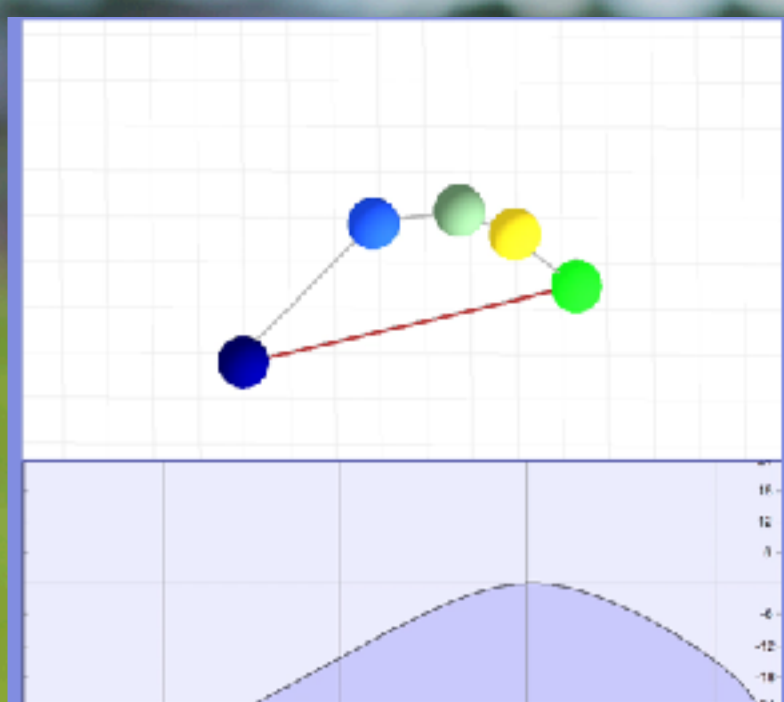
Explicit Gesture Sonification – Deterministic Modelling



Full Upper-Body Tracking with Depth Cameras (Intangible Musical Instrument)

Explicit Gesture Sonification – Deterministic Modelling





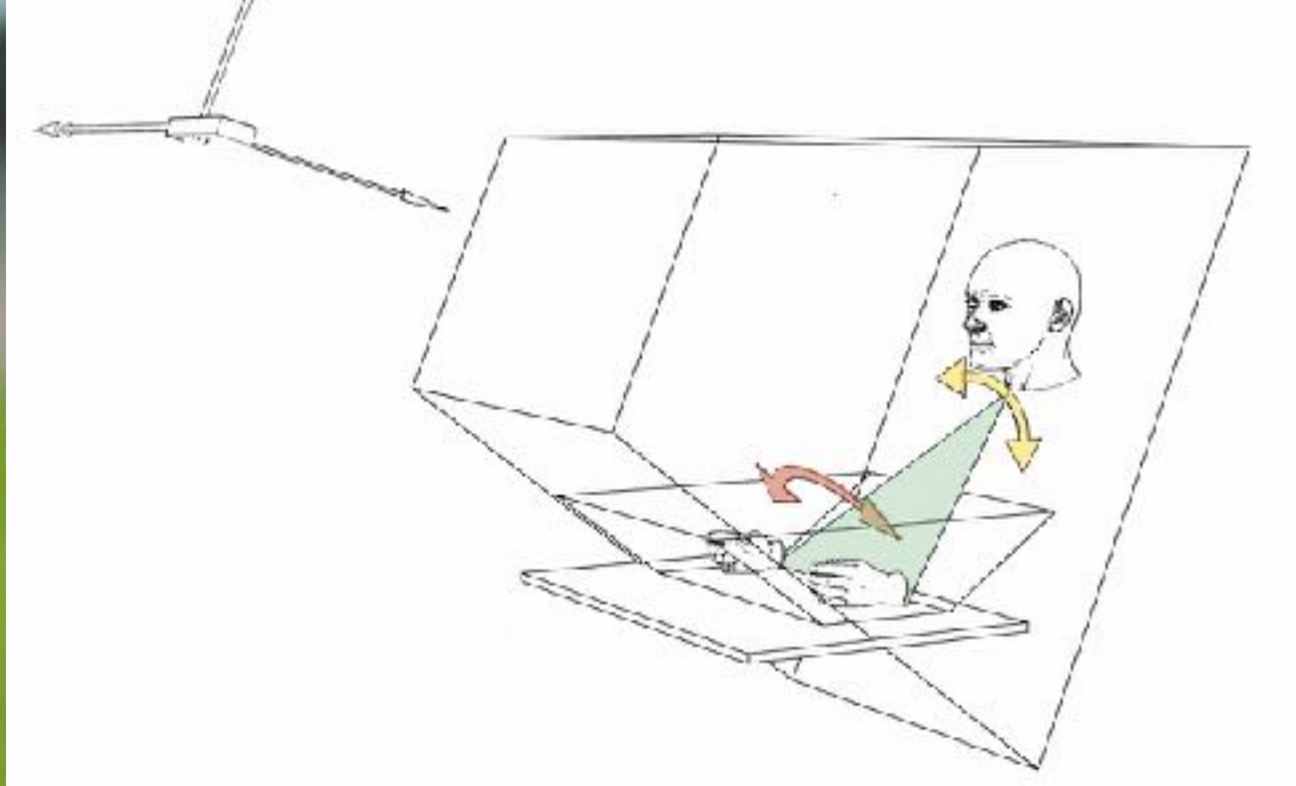
Full Upper-Body Tracking with Depth Cameras (Intangible Musical Instrument)
Explicit Gesture Sonification – Deterministic Modelling

Kite-flying control:

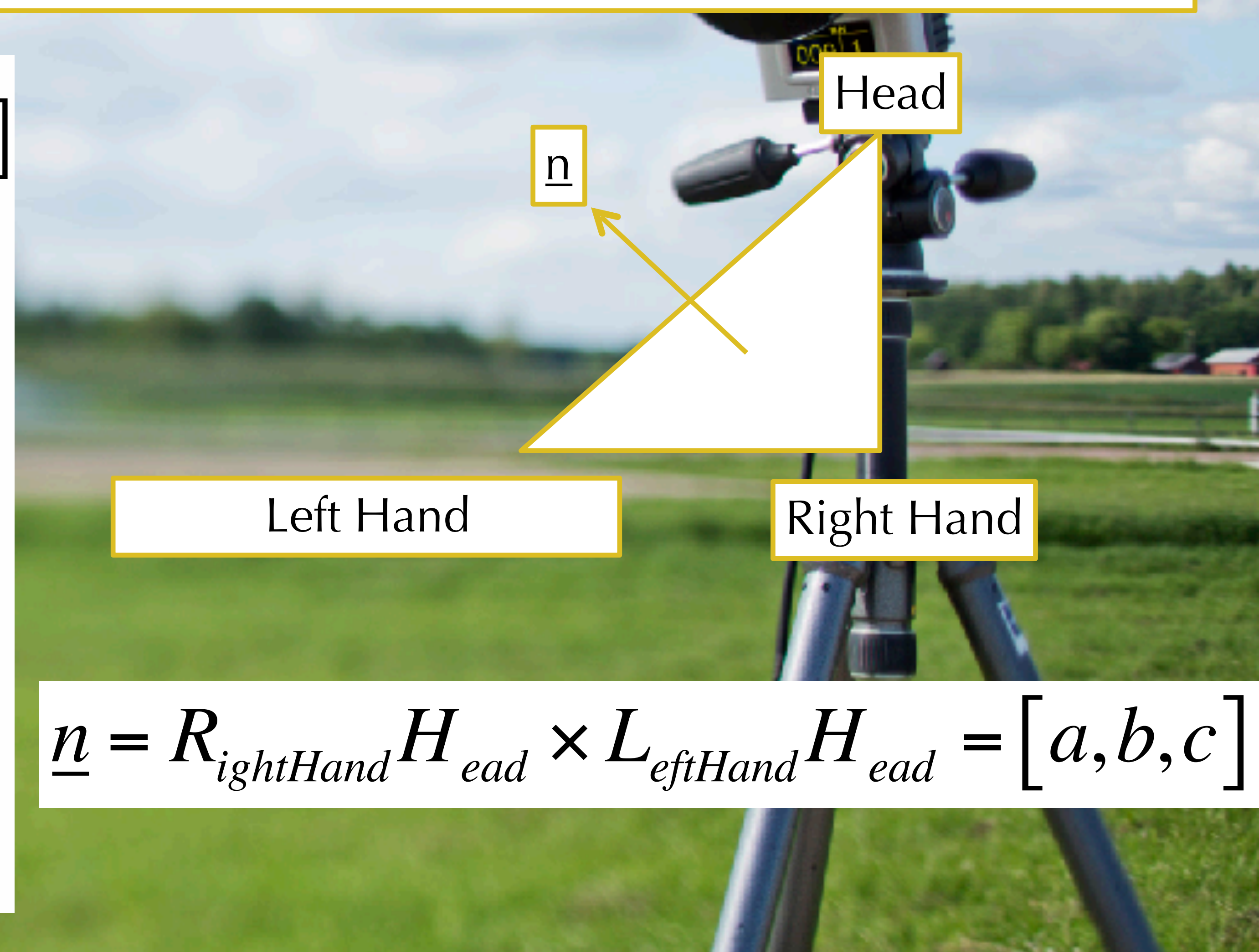
triangle plane′ orientation (green) vs. Kinect′ ***xy*** plane provides a sense of how much left or right your body is rotating (red arrow).

xz vs. ***triangle*** plane reacts if the body is going backward or forward

and/or the hands are going higher or lower (yellow arrow)



$$\underline{n} = R_{ightHand}H_{ead} \times L_{eftHand}H_{ead} = [a,b,c]$$



$$\underline{n} = R_{ightHand}H_{ead} \times L_{eftHand}H_{ead} = [a,b,c]$$

« The future is independent of the past, given the present »

Andreï Andreïevitch Markov
Андрей Андреевич Марков
2 June 1856 - 20 July 1921

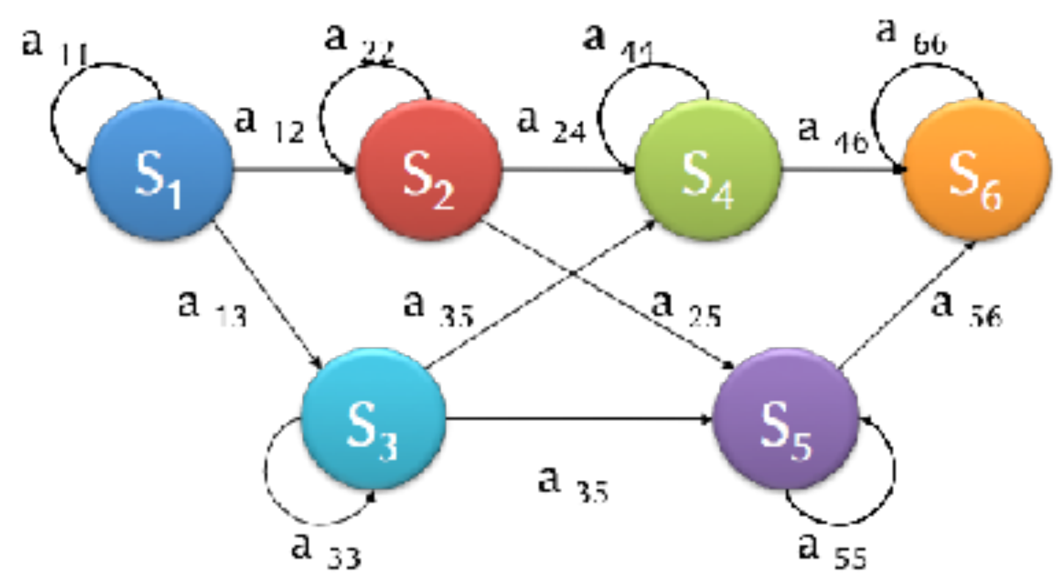
The concept of Hidden Markov Models
Introduction

Credits: Lane Votapka

- We want to reason about a sequence of observations
  - Gesture recognition in Human-Robot Collaboration
  - Visual-speech recognition
  - Gesture control of robots
- Need introduce time or space into our models

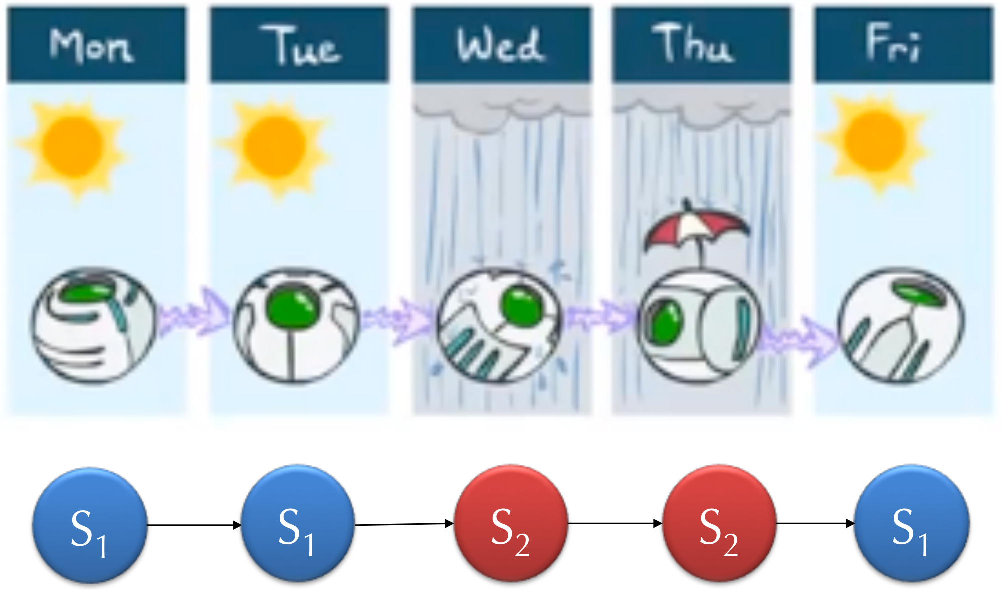
Markov Chains
Model definition

- Set of N States, $\{S_1, S_2, \ldots S_N\}$
- Sequence of states $Q = \{q_1, q_2, \ldots\}$
- Initial probabilities $\pi = \{\pi_1, \pi_2, \ldots \pi_N\}$
  - $\pi_i = P(q_1 = S_i)$
- Transition matrix A NxN
  - $a_{ij} = P(q_{t+1} = S_j \mid q_t = S_i)$



Weather model:

- 3 states {sunny, rainy, cloudy}



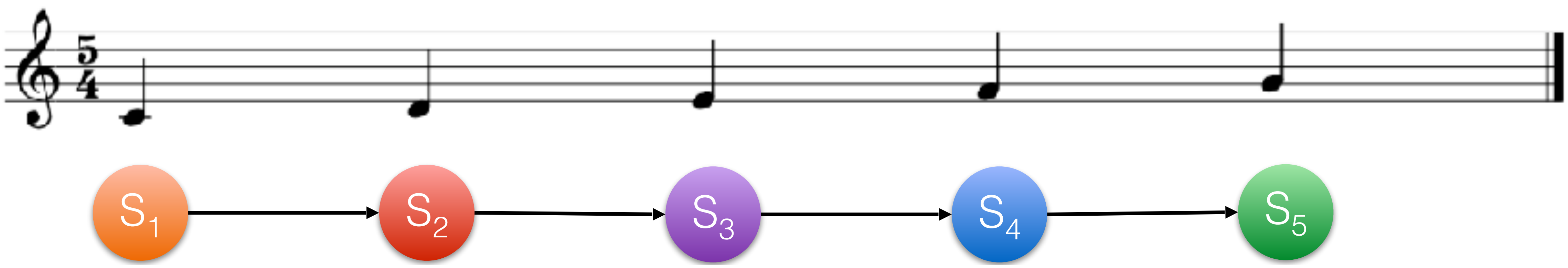
Problem:

- Forecast weather state, based on the current weather state

Markov Chain
Example in musical gestures

Let’s assume a set of 5 musical states, $\{S_1, S_2, S_3, S_4, S_5\}$

S_1 = fingering_1, S_2 = fingering_2, S_3 = fingering_3, S_4 = fingering_4, S_5 = fingering_5



Let’s assume a simplified model of fingering prediction based on the previous fingerings

$$P = (q_n | q_{n-1}, q_{n-2}, \ldots, q_1) \tag{1}$$

We can give the probabilities of fingerings for the next time using n times of history

e.g. if we knew that fingerings for the past three times were $\{S_1, S_2, S_3\}$ in chronological order, the probability that the next fingering will be S_4 is given by:

$$P = (q_4 = S_4 | q_3 = S_3, q_2 = S_2, q_1 = S_1) \tag{2}$$

Markov Chain
Example in musical gestures

But, here's the problem: the larger the n is, the more statistics we need

e.g. for n=5, $5^5 = 3125$ past histories !

Therefore, we will make a simplifying assumption, called the Markov Assumption or Property

$$P(q_n|q_{n-1}, q_{n-2}, \dots, q_1) \approx P(q_n|q_{n-1}) \quad (3)$$

first-order Markov Assumption

a second-order Markov Assumption would have q_n depend on q_{n-1} and q_{n-2}

So, we can express the joint probability using the Markov Assumption

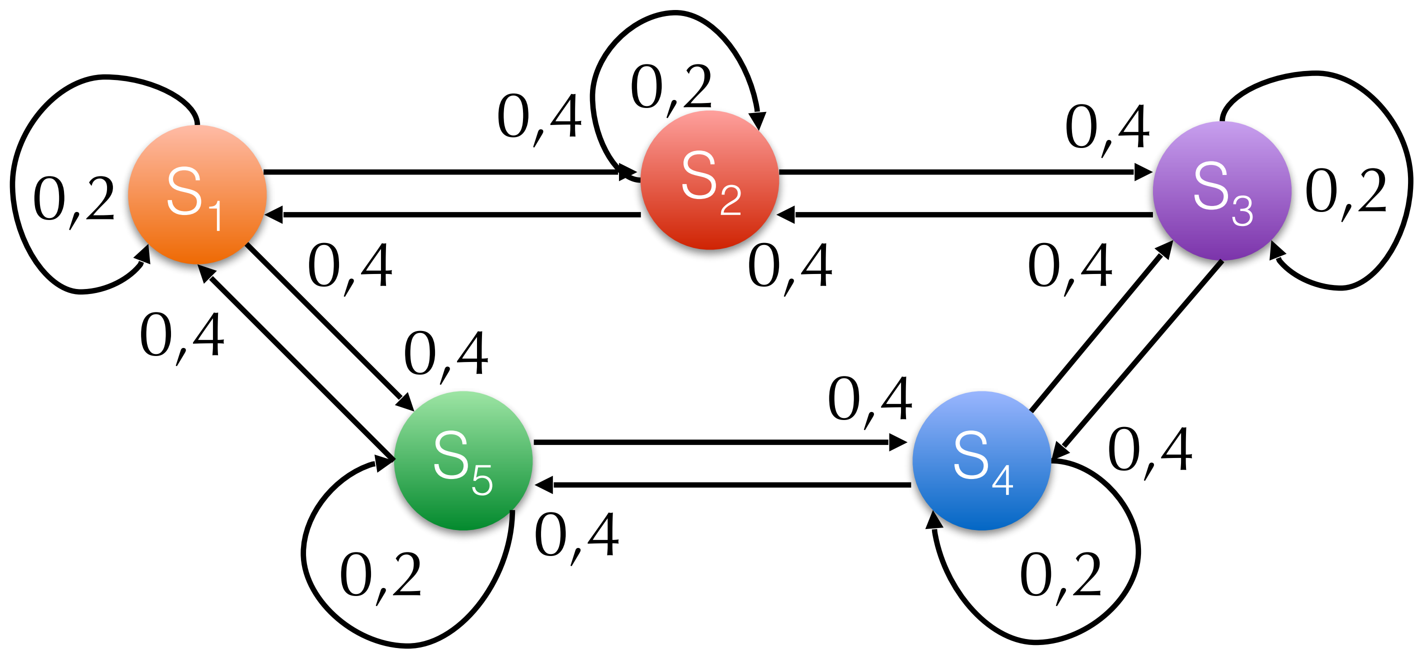
$$P(q_1, \dots, q_n) = \prod_{i=1}^{n} P(q_i|q_{i-1}) \quad (4)$$

Hence, this now has a profound affect on the number of histories that we have to find

we only need $5^2 = 25$ past histories to characterize the probabilities of all histories !

Markov Chain
Example in musical gestures

Let's pick arbitrarily some numbers for $P(q_i|q_{i-1})$ and draw a probabilistic finite state automaton



Question 1

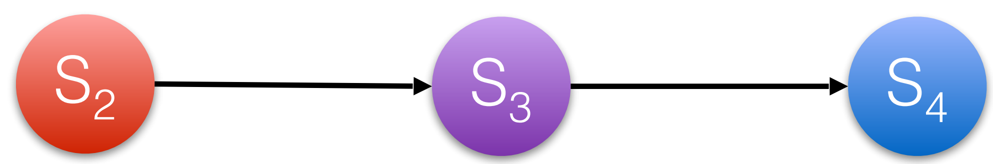
Given that now the performer is playing an S_2, what's the probability that his/her next fingering is an S_3 and the fingering after is an S_4?

Question 2

Given that now the performer is playing an S_2, what's the probability that s/he will be playing an S_4 in three fingerings from now?

Markov Chain
Example in musical gestures

Question 1



This translates into:

$$
\begin{aligned}
P\left(q_2 = S_3, q_3 = S_4 \middle| q_1 = S_2\right) &= P(q_3 = S_4 | q_2 = S_3, q_1 = S_2) * \\
&\quad P(q_2 = S_3 | q_1 = S_2) \\
&= P(q_3 = S_4 | q_2 = S_3) * \\
&\quad P(q_2 = S_3 | q_1 = S_2) \\
&= 0{,}4 * 0{,}4 \\
&= 0{,}16
\end{aligned}
$$

You can also think this as moving through the automaton, multiplying the probabilities

Markov Chain

Example in musical gestures

Question 2

This translates into:

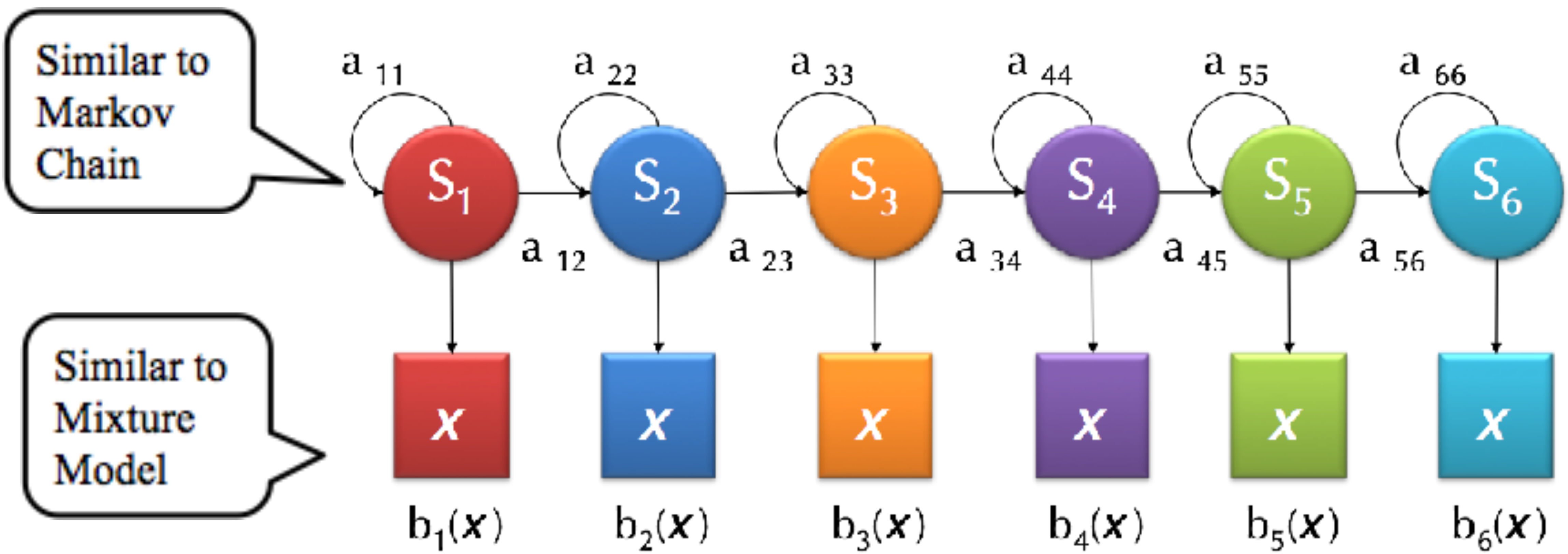
$$
\begin{aligned}
P(q_4 = S_4 | q_1 = S_2) &= P(q_3 = S_3, q_2 = S_3 | q_1 = S_2) + \\
&\quad P(q_3 = S_4, q_2 = S_3 | q_1 = S_2) \\
&= P(q_3 = S_3 | q_2 = S_3) * P(q_2 = S_3 | q_1 = S_2) + \\
&\quad P(q_3 = S_4 | q_2 = S_3) * P(q_2 = S_3 | q_1 = S_2) \\
&= 0{,}2 * 0{,}4 + 0{,}4 * 0{,}4 \\
&= 0{,}24
\end{aligned}
$$

we need observations to update our beliefs

$\lambda=(A, B, \pi)$: Hidden Markov Model

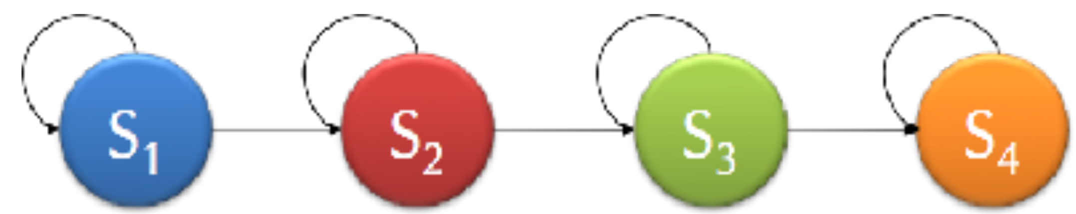
- $A=\{a_{ij}\}$: Transition probabilistic distribution
  - $a_{ij}=P(q_{t+1}=S_j \mid q_t=S_i)$ Hidden
- $B=\{b_i(\mathbf{x})\}$: Emission probabilistic distribution
  - $b_i(\mathbf{O}_t)=P(\mathbf{O}_t=\mathbf{x} \mid q_t=S_i)$ Observed
- $\pi=\{\pi_i\}$: Initial state probabilistic distribution
  - $\pi_i=P(q_1=S_i)$

- Basic conditional independence:
  - Past and future are independent of the present
  - Each time step only depends on the previous
  - This is called the first order Markov property

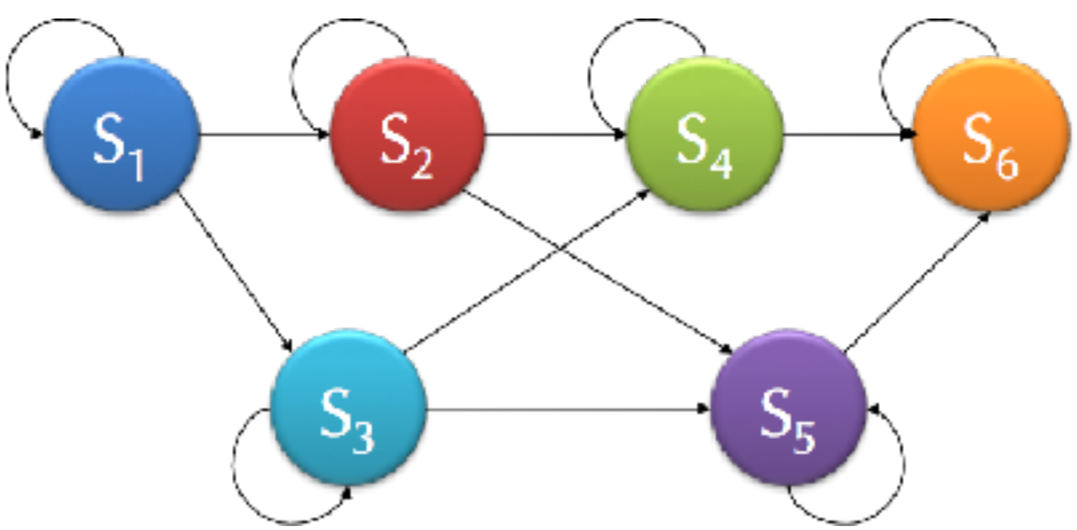

Similar to Markov Chain
Similar to Mixture Model
a 11
a 22
a 33
a 44
a 55
a 66
S1
S2
S3
S4
S5
S6
a 12
a 23
a 34
a 45
a 56
x
x
x
x
x
x
b1(x)
b2(x)
b3(x)
b4(x)
b5(x)
b6(x)

Left to right (A)

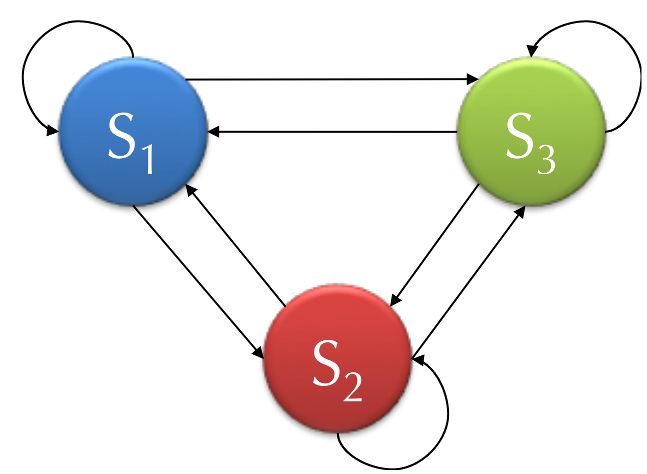
Left to right (B)



Left to right (C)



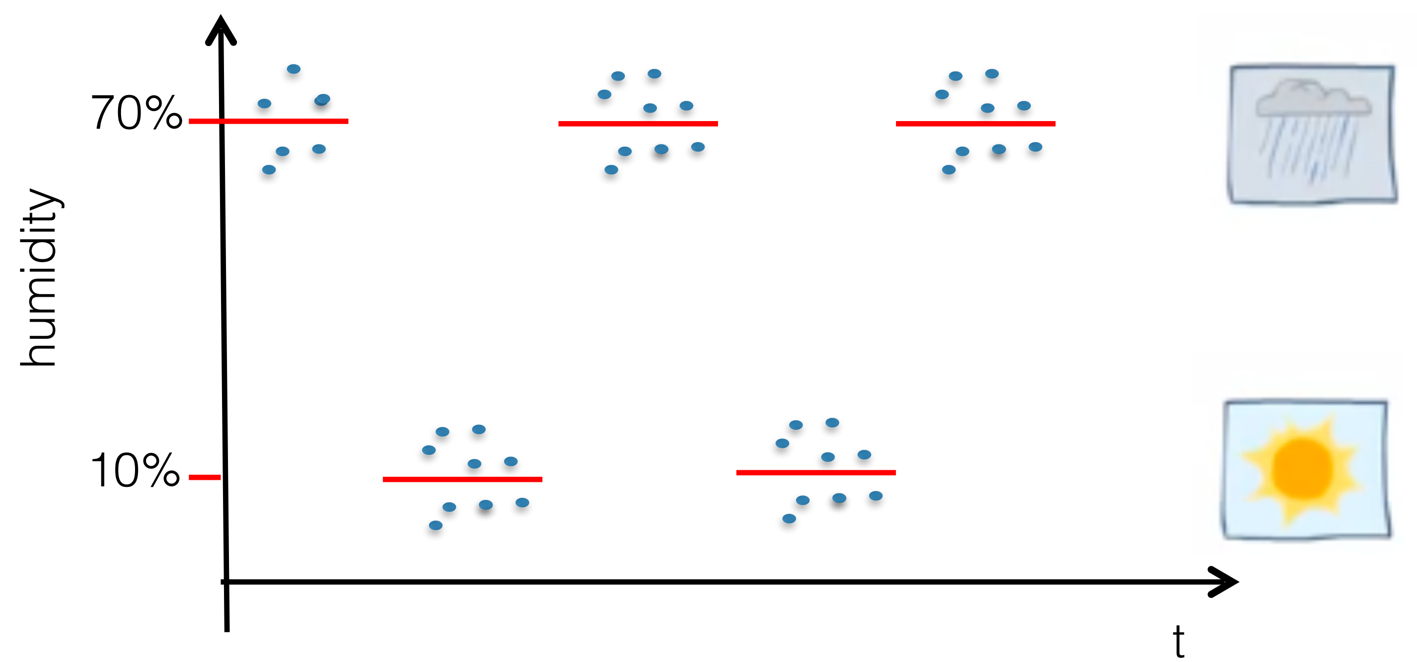
Ergodic



Weather model:
- 2 “hidden” states
  - {rainy, cloudy}

- Measure weather-related variables
  (e.g. humidity)

Problem:
Forecast the weather state, given the current weather variables



$x \epsilon \mathbb{R}$ $S_i \epsilon \{rainy, cloudy\}$ $A = \begin{bmatrix} 0,99 & 0,1 \\ 0,2 & 0,98 \end{bmatrix}$

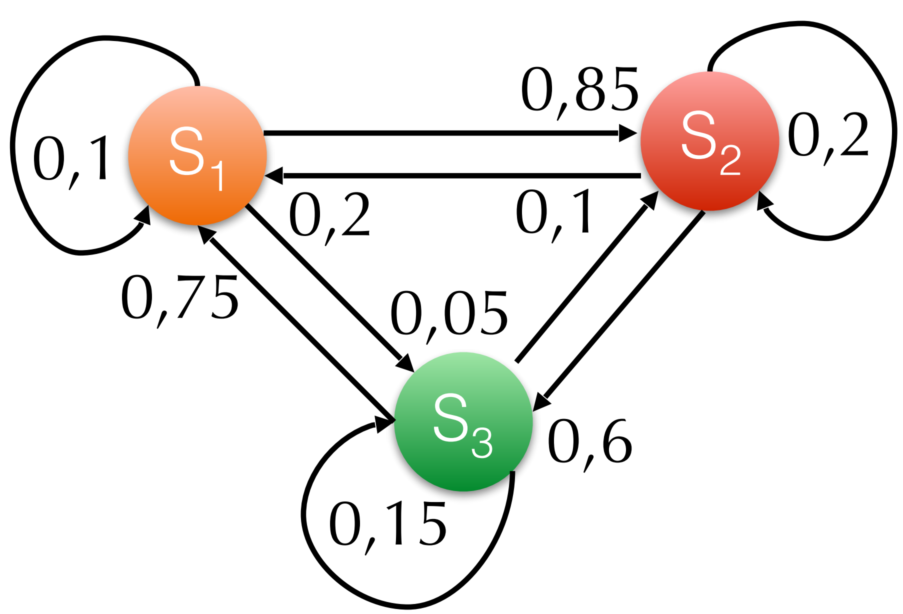
Hidden Markov Model
Example in human-robot collaboration

Suppose that you want to program a robot to provide to the worker with components for assembling motor hoses

The only input to the robot is whether there are available components in the box or not

The possible states of technical gestures of the worker are:

S_1 = Take two components, S_2 = Join the components, S_3 = Screw the components



	Probability of having available components in the box
Take	0,8
Join	0,1
Screw	0,4

Hidden Markov Model
Example in human-robot collaboration

We have to factor the fact that the actual gestures of the worker are "hidden" for the robot

Thus, we will be based on the Bayes' Rule

$$P(q_1, \ldots, q_n | O_1, \ldots, O_n) = \frac{P(O_1, \ldots, O_n | q_1, \ldots, q_n) P(q_1, \ldots, q_n)}{P(O_1, \ldots, O_n)} \quad (5)$$

where O_i is true if the box has a component inside at the moment *i* and false if not

The probability $P(O_1, \ldots, O_n | q_1, \ldots, q_n)$ **can be estimated as**

$$P(O_1, \ldots, O_n | q_1, \ldots, q_n) = \prod_{i=1}^{n} P(O_i | q_i) \quad (6)$$

If we assume that for all i, given q_i, O_i is independent of all O_j and q_j for all $j \neq i$

Hidden Markov Model
Example in human-robot collaboration

Question 1

Suppose that the worker is currently joining the components and at the next time stamp, there were available components into the box. Assuming that the prior probability of having available components on the box at any time is 0,5, what's the probability that at the next time stamp the worker was screwing the components?

$$P(q_2 = S_3 | q_1 = S_2, b_2 = True) = \frac{P(q_2{=}S_3, q_1{=}S_2 | b_2{=}\mathrm{T})}{P(q_1{=}S_2 | b_2{=}\mathrm{T})}$$

$$= \frac{P(q_2{=}S_3, q_1{=}S_2 | b_2{=}\mathrm{T})}{P(q_1{=}S_2)}$$

$$= \frac{P(b_2{=}\mathrm{T} | q_1{=}S_2, q_2{=}S_3 |) P(q_2{=}S_3, q_1{=}S_2)}{P(q_1{=}S_2) P(b_2{=}\mathrm{T})}$$

$$= \frac{P(b_2{=}\mathrm{T} | q_2{=}S_3) P(q_2{=}S_3, q_1{=}S_2)}{P(q_1{=}S_2) P(b_2{=}\mathrm{T})}$$

$$= \frac{P(b_2{=}\mathrm{T} | q_2{=}S_3) P(q_2{=}S_3 | q_1{=}S_2) P(q_1{=}S_2)}{P(q_1{=}S_2) P(b_2{=}\mathrm{T})}$$

Hidden Markov Model
Example in human-robot collaboration

$$(Cancel: P(q_1 = S_2)) = \frac{P(b_2{=}\mathrm{T}|q_2{=}S_3|)P(q_2{=}S_3|q_1{=}S_2)}{P(q_1{=}S_2)P(b_2{=}\mathrm{T})}$$

$$= \frac{0{,}4*0{,}6}{0{,}5}$$

$$= 0{,}48$$

Hidden Markov Model
Example in human-robot collaboration

Question 2

Suppose that the worker is currently joining the components while there were available components into the box in the time stamp 2 but not in the time stamp 3. Assuming that the prior probability of having available components on the box at any time is 0,5, what's the probability that at the time stamp 3 the worker was screwing the components?

$$P(q_3 = S_3 | q_1 = S_2, b_2 = True, b_3 = False) =$$

$$P(q_2 = S_3, q_3 = S_3 | q_1 = S_2, b_2 = T, b_3 = F) + P(q_2 = S_2, q_3 = S_3 | q_1 = S_2, b_2 = T, b_3 = F) +$$
$$P(q_2 = S_1, q_3 = S_3 | q_1 = S_2, b_2 = T, b_3 = F) =$$

$$\frac{P(b_3 = F | q_3 = S_3) P(b_2 = T | q_2 = S_3) P\left(q_3 = S_3 | q_2 = S_3\right) P\left(q_2 = S_3 | q_1 = S_2\right) P(q_1 = S_2)}{P(b_3 = F) P(b_2 = T) P(q_1 = S_2)} +$$

$$\frac{P(b_3 = F | q_3 = S_3) P(b_2 = T | q_2 = S_2) P\left(q_3 = S_3 | q_2 = S_2\right) P\left(q_2 = S_2 | q_1 = S_2\right) P(q_1 = S_2)}{P(b_3 = F) P(b_2 = T) P(q_1 = S_2)} +$$

Hidden Markov Model
Example in human-robot collaboration

$$\frac{P(b_3 = F|q_3 = S_3)P(b_2 = T|q_2 = S_1)P\left(q_3 = S_3|q_2 = S_1\right)P\left(q_2 = S_1|q_1 = S_2\right)P(q_1 = S_2)}{P(b_3 = F)P(b_2 = T)P(q_1 = S_2)} =$$

$$\frac{P(b_3 = F|q_3 = S_3)P(b_2 = T|q_2 = S_3)P\left(q_3 = S_3|q_2 = S_3\right)P\left(q_2 = S_3|q_1 = S_2\right)}{P(b_3 = F)P(b_2 = T)} +$$

$$\frac{P(b_3 = F|q_3 = S_3)P(b_2 = T|q_2 = S_2)P\left(q_3 = S_3|q_2 = S_2\right)P\left(q_2 = S_2|q_1 = S_2\right)}{P(b_3 = F)P(b_2 = T)} +$$

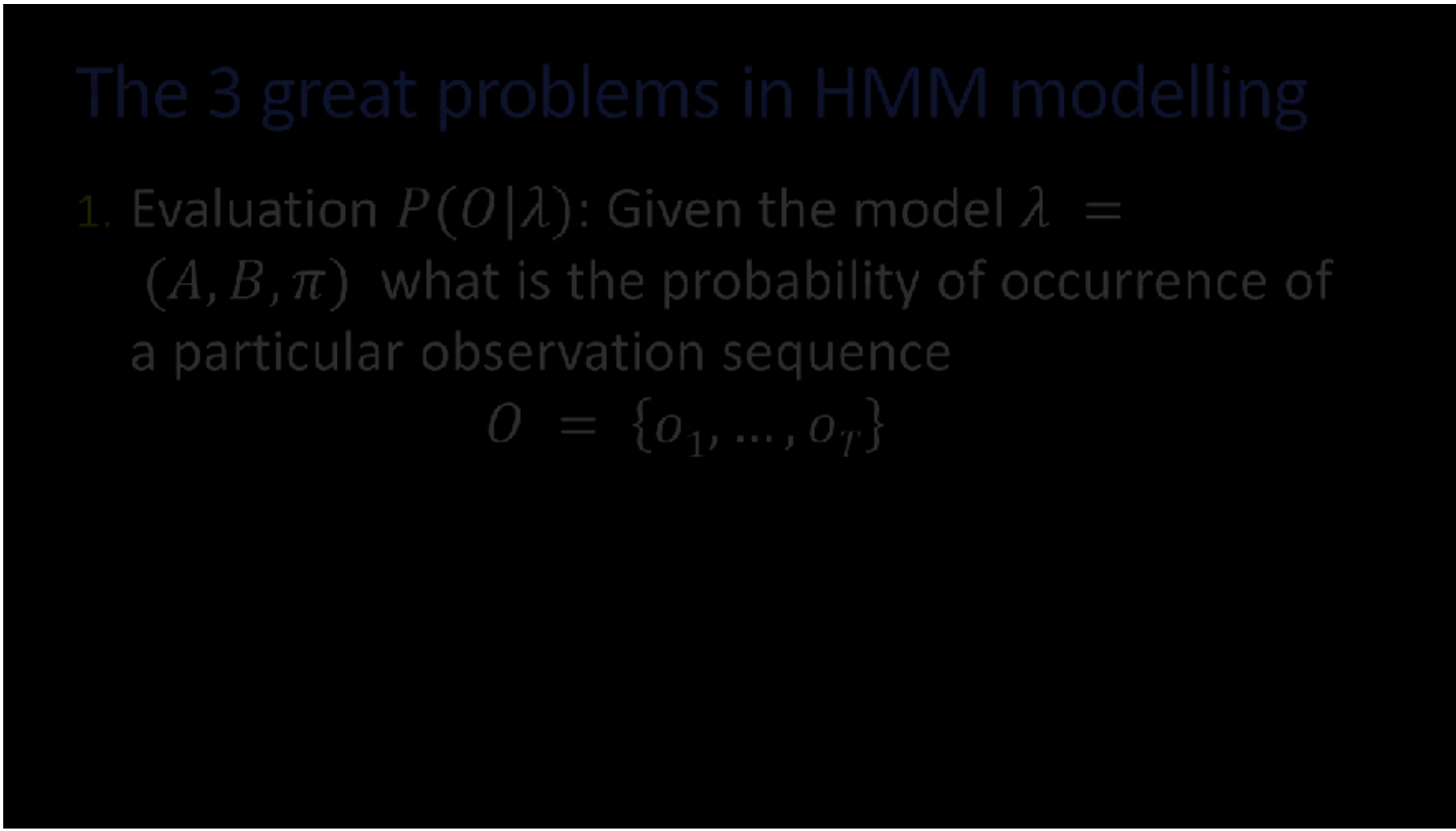
$$\frac{P(b_3 = F|q_3 = S_3)P(b_2 = T|q_2 = S_1)P(q_3 = S_3|q_2 = S_1)P(q_2 = S_1|q_1 = S_2)}{P(b_3 = F)P(b_2 = T)P(q_1 = S_2)} =$$

$$\frac{(0,6)(0,4)(0,15)(0,6)}{(0,5)(0,5)} + \frac{(0,6)(0,1)(0,6)(0,2)}{(0,5)(0,5)} + \frac{(0.6)(0.8)(0,05)(0,2)}{(0,5)(0,5)} =$$

$$\frac{(0,6)(0,4)(0,15)(0,6)}{(0,5)(0,5)} + \frac{(0,6)(0,1)(0,6)(0,2)}{(0,5)(0,5)} + \frac{(0,6)(0,8)(0,05)(0,2)}{(0,5)(0,5)} =$$

$$0.0864 + 0.02888 + 0.0192 = 0,13448$$

- Evaluation
  - $O, \lambda \rightarrow P(O|\lambda)$
- Uncover the hidden part
  - $O, \lambda \rightarrow$ Q that $P(Q|O, \lambda)$ is maximum
- Learning
  - $\{O\} \rightarrow \lambda$ that $P(O|\lambda)$ is maximum



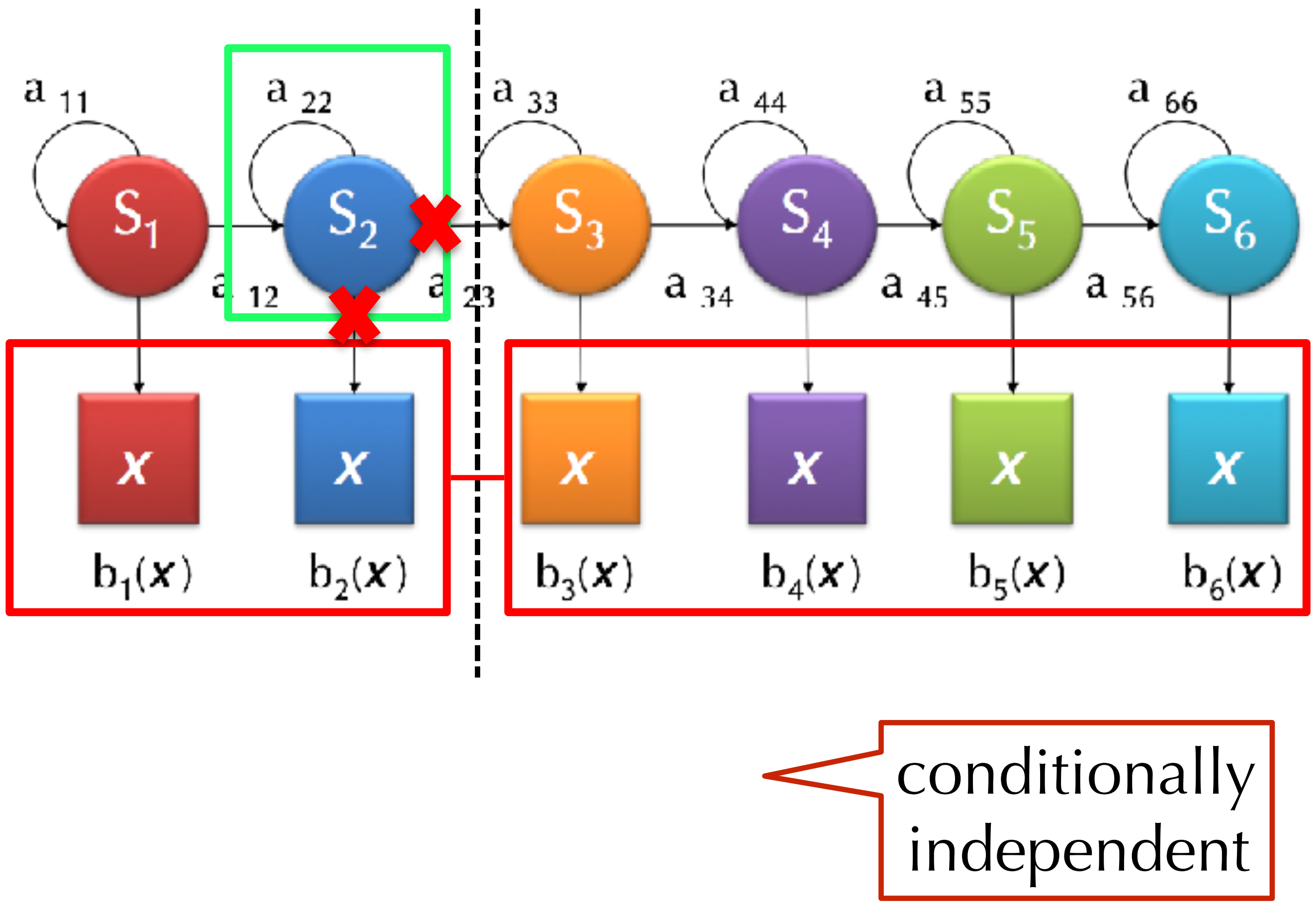
Credits: Aaron Bobick

$O, \lambda \rightarrow P(O|\lambda)$

- Solved by the **Forward** algorithm

Applications

- Find some likely samples
- Evaluation of a sequence of observations
- Change detection



Initialisation

$$\alpha_1(i) = \pi_i * b_i(o_1)$$

Induction

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] * b_j(o_{t+1})$$

Termination

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

Basic problems – Uncover the hidden path

O, λ → Q that P(Q|O, λ) is maximum

- Solved by **Viterbi** algorithm
- No « correct » sequence to be found

How to solve it:

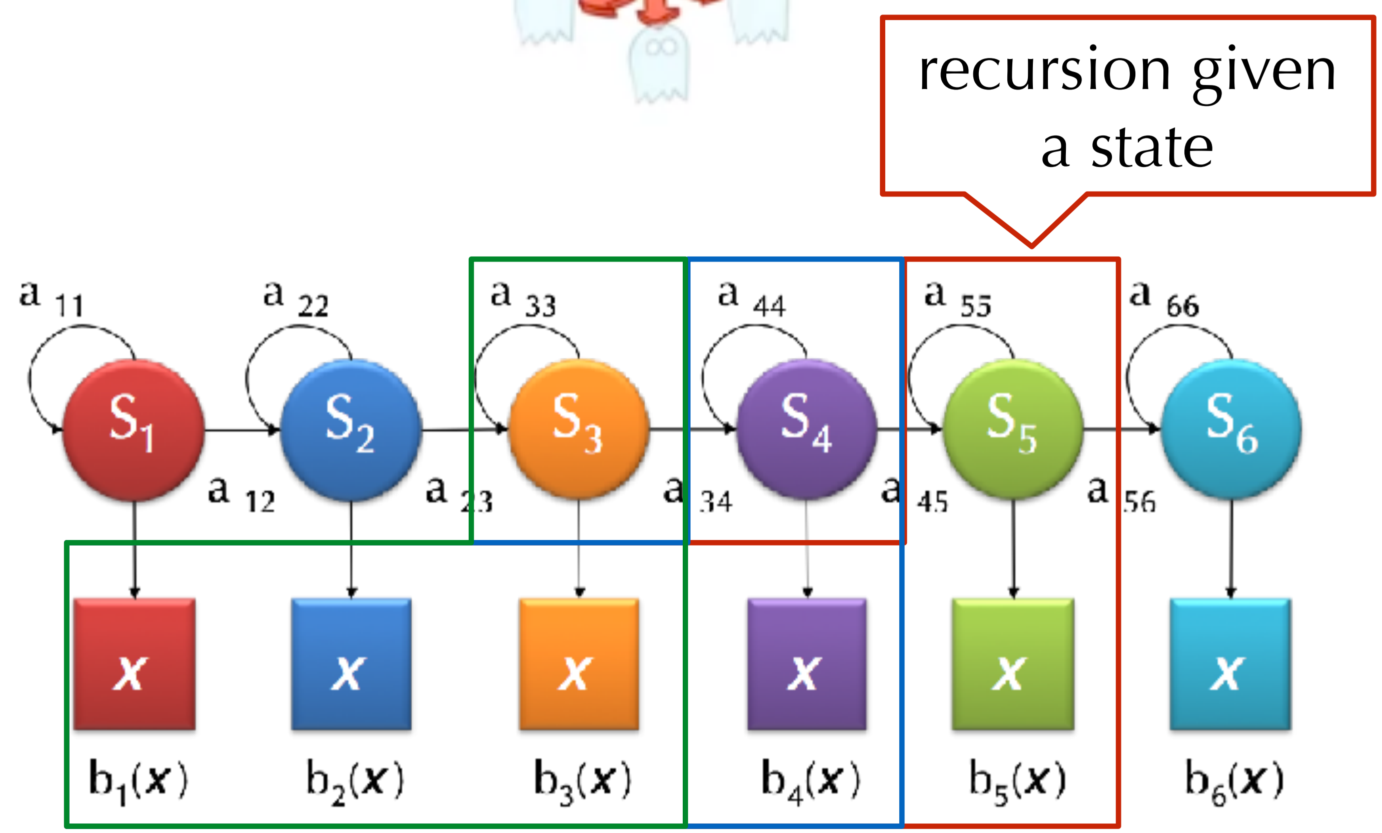
- Use an optimality criterion that depends on the use of the uncovered state sequence

Possible uses:

- Learn about the structure of the model
- Get average statistics of the states

Applications

- Find the real states by maximising the likelihood until a given state
- Find some recursion given an arbitrary state
- Used in the learning problem



Initialisation

$$\delta_1(i) = \pi_i * b_i(o_1)$$

Induction

$$\delta_t(j) = \max_{1\le i\le N}[\delta_{t-1}(i) * a_{ij}]b_j(O_t)\,;\; \psi_t(j) = \arg\max_{1\le i\le N}[\delta_{t-1}(i) * a_{ij}]$$

Termination

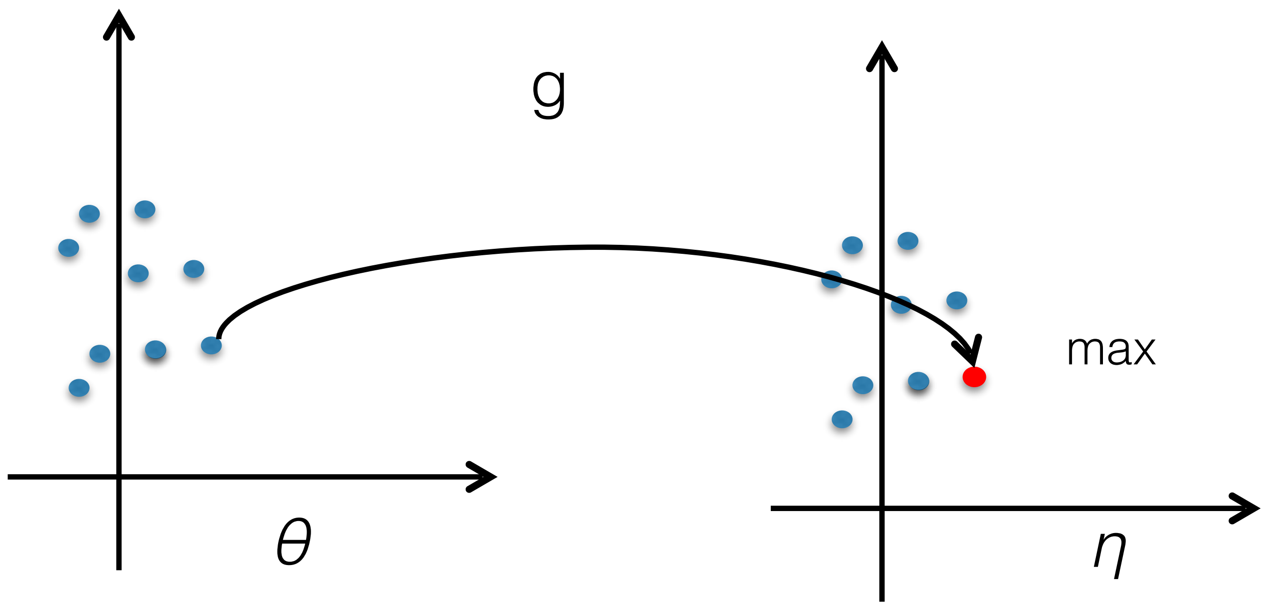
$$P^* = \max_{1\le i\le N}[\delta_T(i)]; \quad q_T^* = \arg\max_{1\le i\le N}[\delta_{t-1}(i) * a_{ij}]$$

Backtracking

$$q_t^* = \psi_t(q_{t+1}^*) \text{ pour } t = T-1 \text{ to } 1$$

- {O} → λ that P(O|λ) is maximum
- No analytic solution
- Solved by **Baum-Welch** (EM variation) when some data is missing (the states)

- Applications
  - Unsupervised Learning (single HMM)
  - Supervised Learning (multiple HMM)



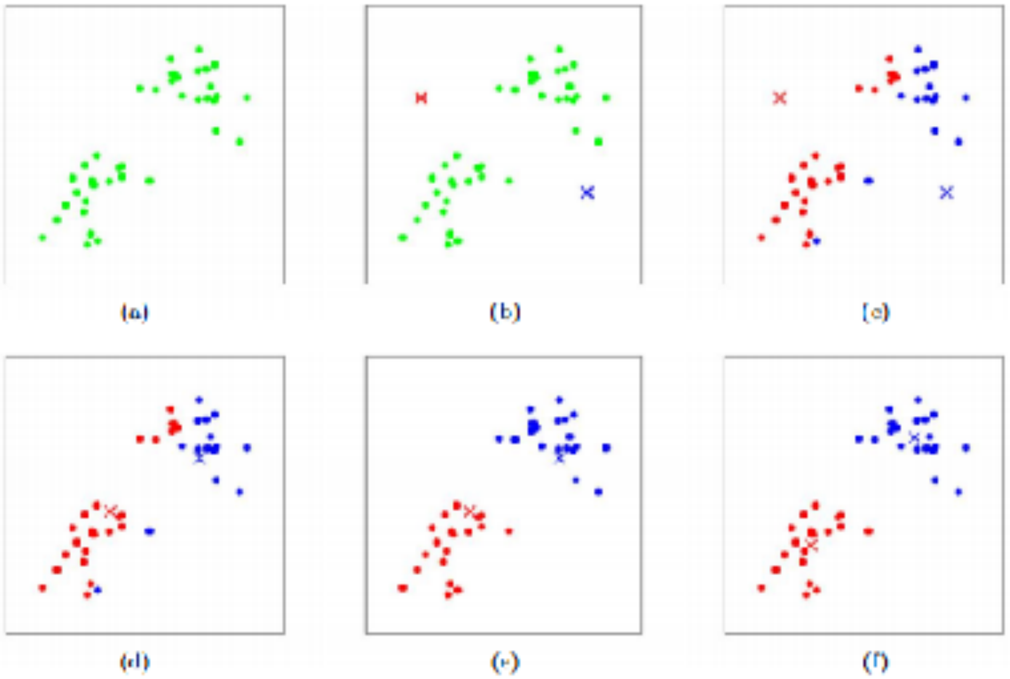
K-Means is an Euclidean-based clustering algorithm

Select initial centroids at random

Assign each object to the cluster with the nearest centroid

Compute each centroid as the mean of the objects assigned to it

Repeat previous 2 steps until no change



Weather model:

- 3 "hidden" states
  - {rainy, cloudy, sunny}

Hidden

- Measure weather-related variables
  (e.g. temperature, humidity, barometric pressure)

Observed

Problem:

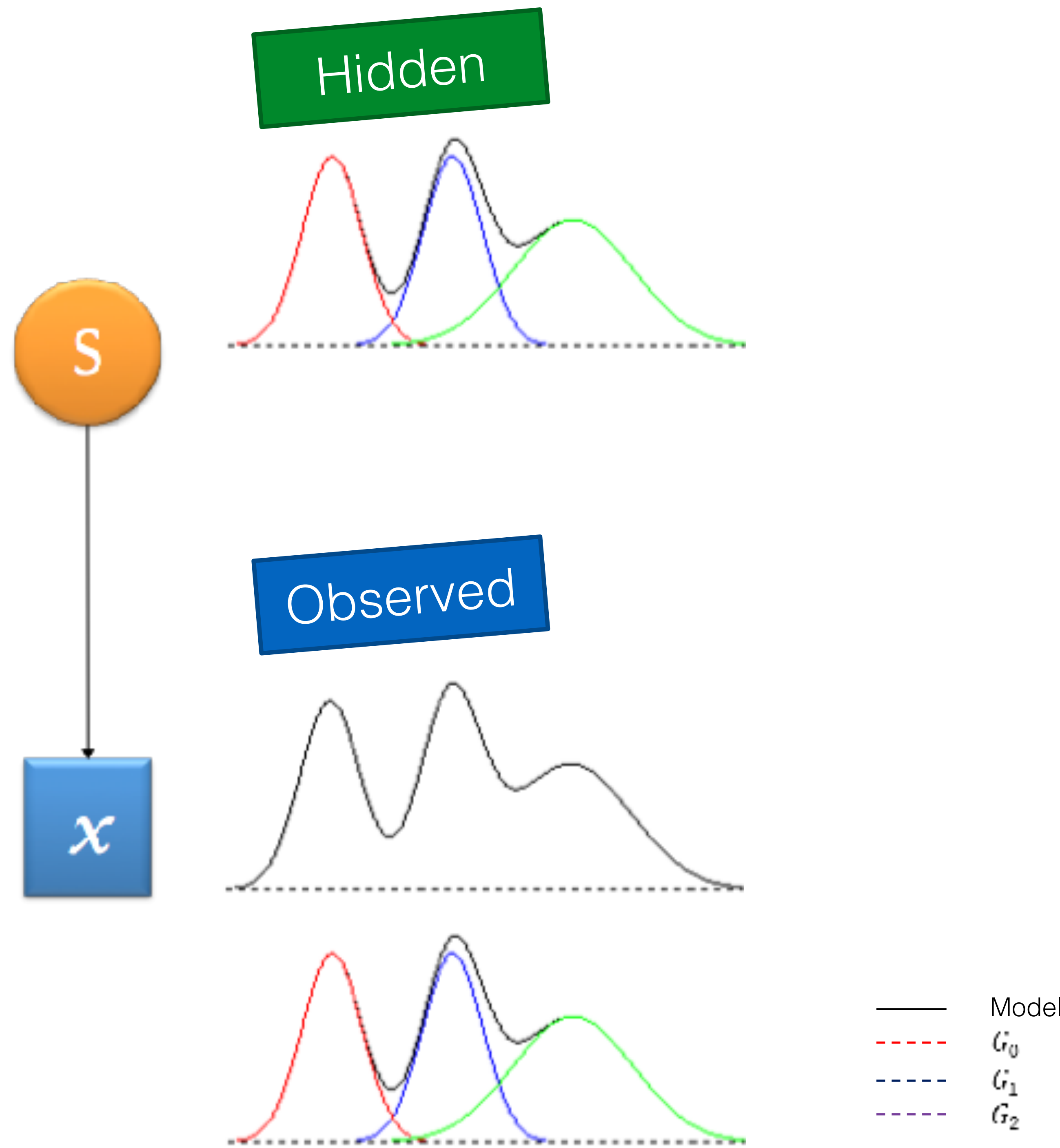
- Given the values of the weather variables, what is the state?

Gaussian Mixture Model
Model definition

- n states observed through an observation **x**

- Model parameter
  $\Theta=\{\Theta_1, \Theta_2.., \Theta_n\}$

$$Model_\Theta = \sum_{i=0}^{n} G_i(\mu_i,\sigma_i)$$



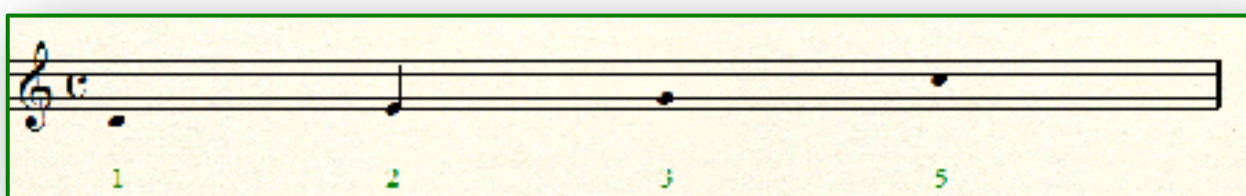
Example in Gesture Recognition
Case study

- Let's consider a gesture dictionnary GD with the following gestures: $GD = \{G_i\}, i\epsilon[1,4]$

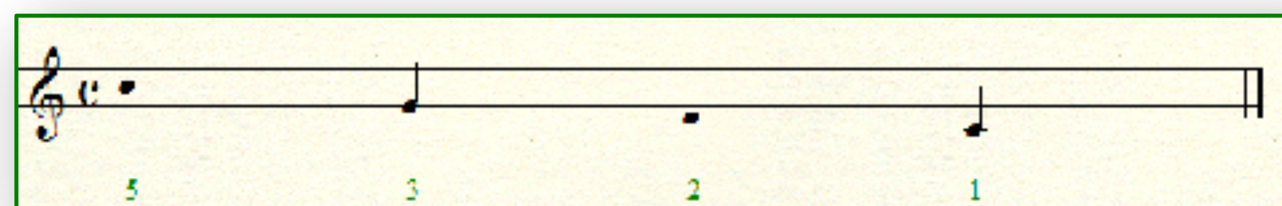
ascending scale

descending scale

ascending arpeggio



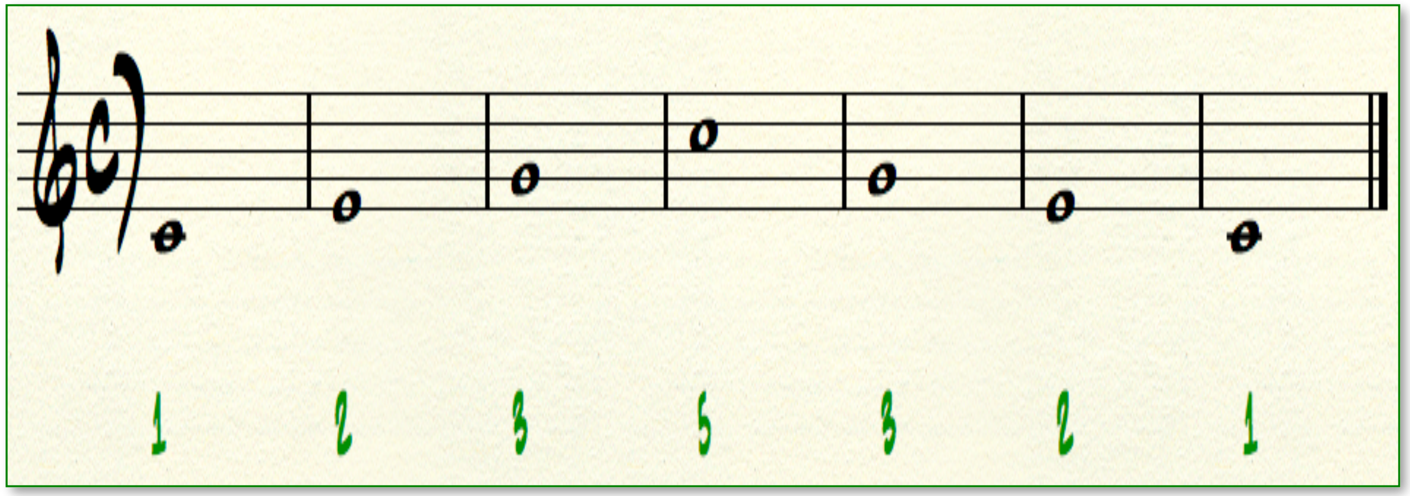
descending arpeggio



- A set of ergodic HMMs, one per gesture: $MD = \{M_i\}, i\epsilon[1,4]$
- The parameters $\lambda_i = (A_i\,, B_i\,, \pi_i\,)$ of all the HMMs

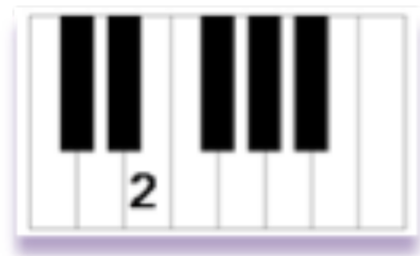
Example in Gesture Recognition
What to recognize

- We want to recognize

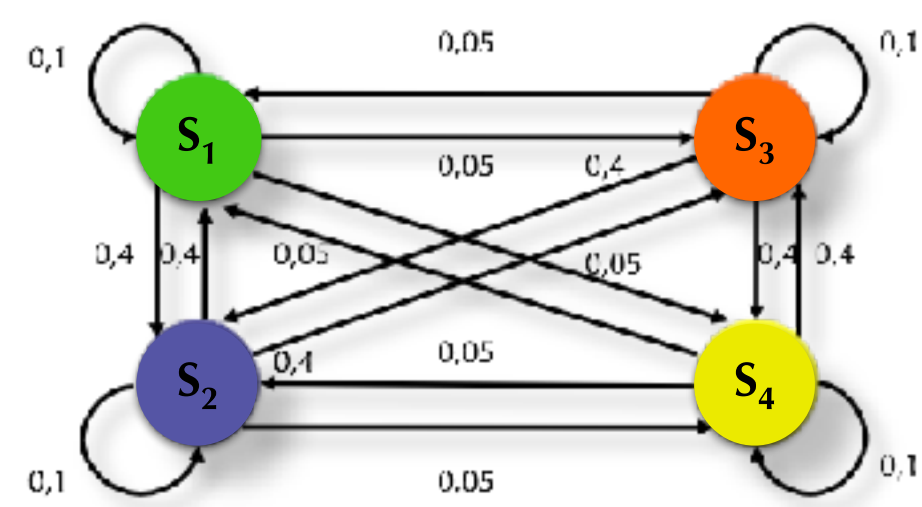


- It is an ascending arpeggio with its inversion

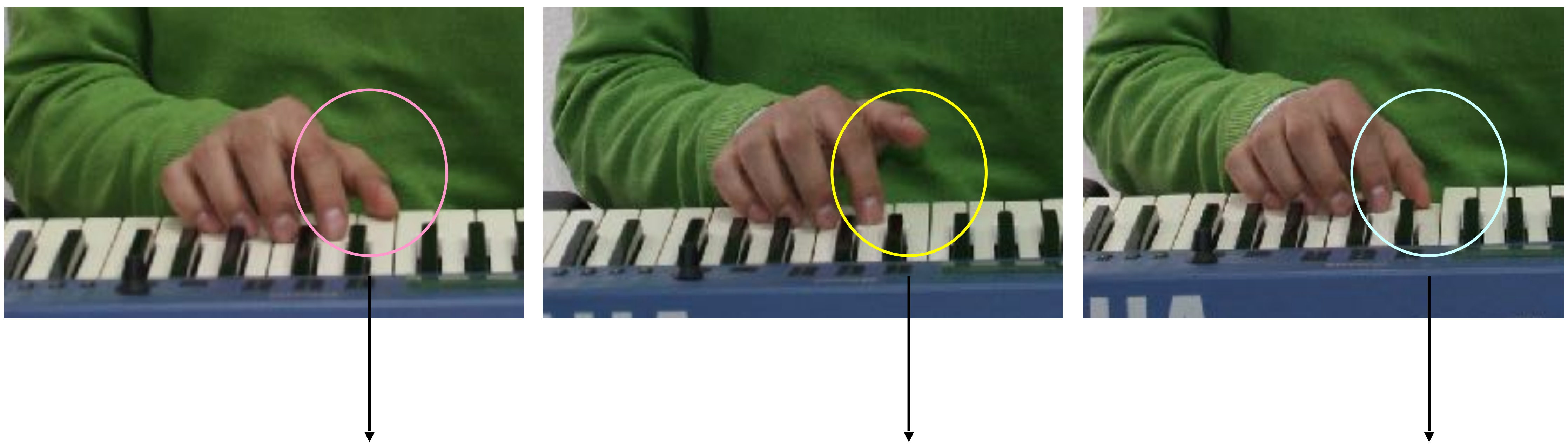
- We consider an alphabet of fingerings

State S_1 DO with 1st fingering	**State S_2** MI with 2nd fingering
	
State S_3 SOL with 3rd fingering	**State S_4** DO with 5th fingering
	

- We assume:
- $A=\{a_{ij}\}$ and
- That $Q=\{q_1, q_2, q_3, q_4, q_5, q_6, q_7\}$ constitutes the ascending arpeggio with its inversion
- $\pi_1=P(q_1)=1$

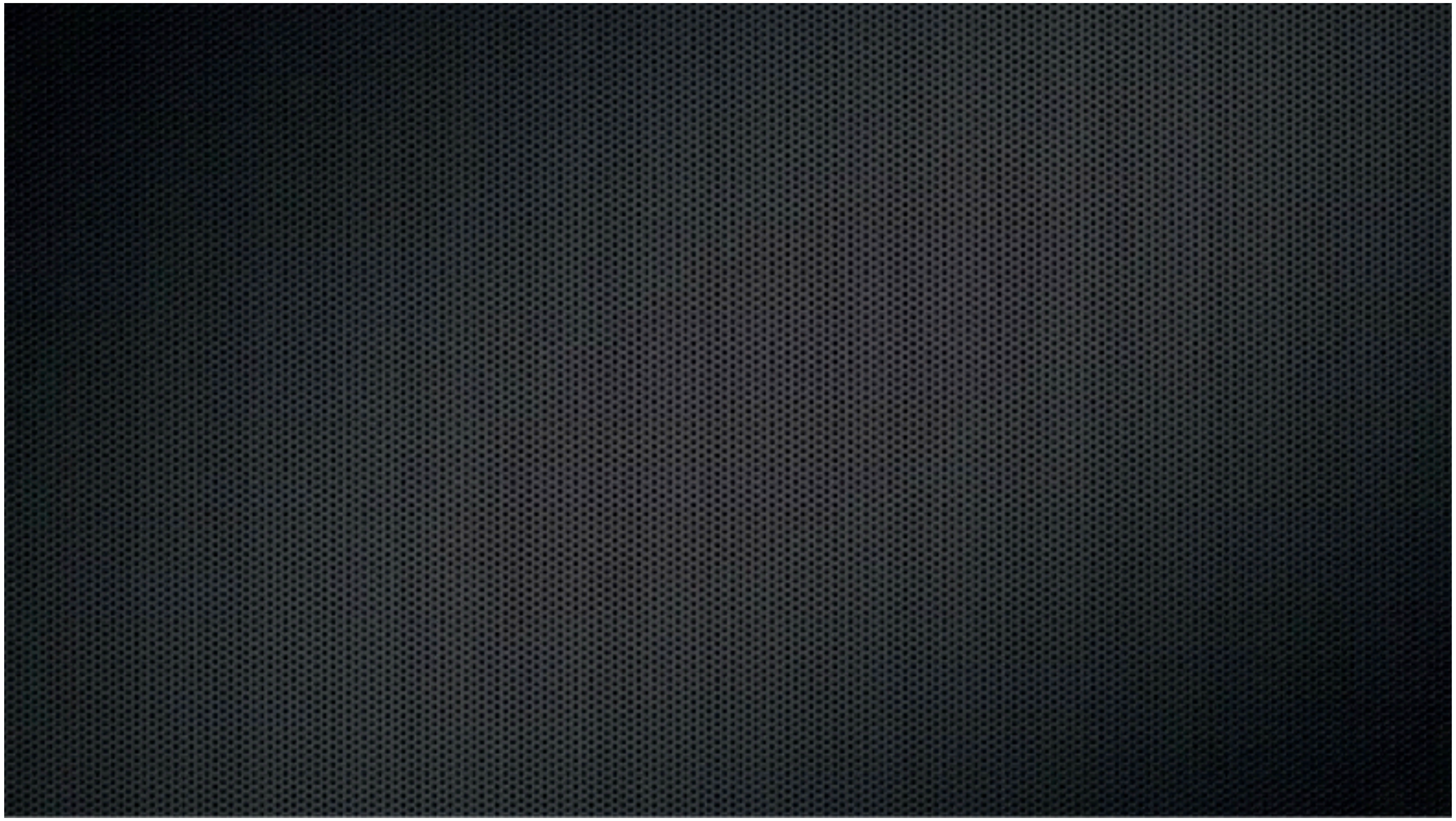


Other modeling could lead to a better physical meaning?



Rest state Start state Attack state

With Gaussian distributions. How many for M_3?



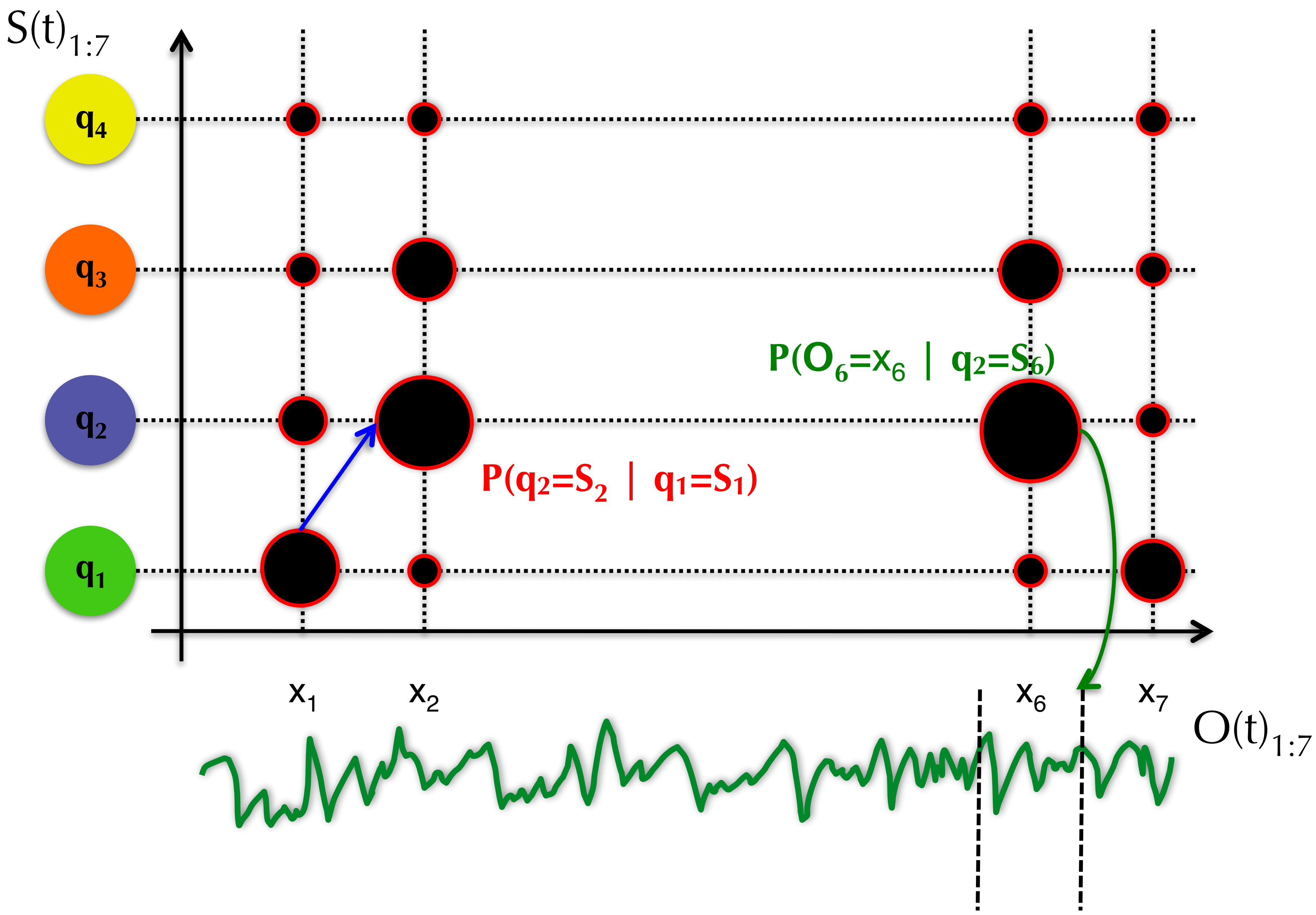
- That the sequence of observations $O(t)_{1:7}$ (visible sequence) is the following:

$$O(t)_{1:7} = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$$

- We assume that M_3 has the maximum likelihood since it is the only ergodic model
- That $S(t)_{1:7}$ is the state sequence (hidden sequence) that generated $O(t)_{1:7}$:

$$Q(t)_{1:7} = \{q_1, q_2, q_3, q_4, q_5, q_6, q_7\}$$

Example in Gesture Recognition
How to represent the model



We know:

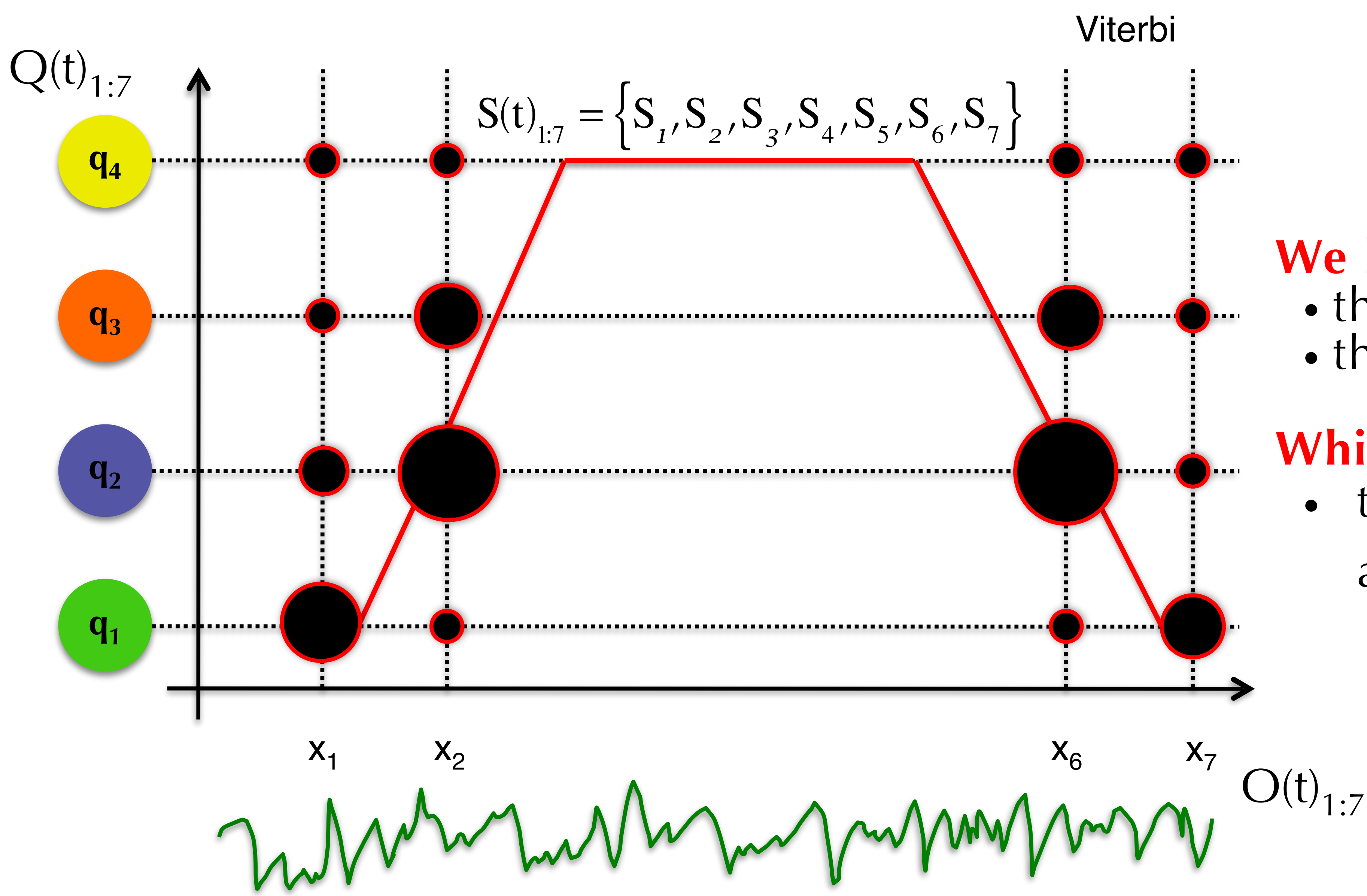
- the model M_3
- the sequence $O(t)_{1:7}$

Which are:

- the λ=(A, B, π) of M_3 that maximize $P(O|\lambda)$

Example in Gesture Recognition

How to uncover the hidden path



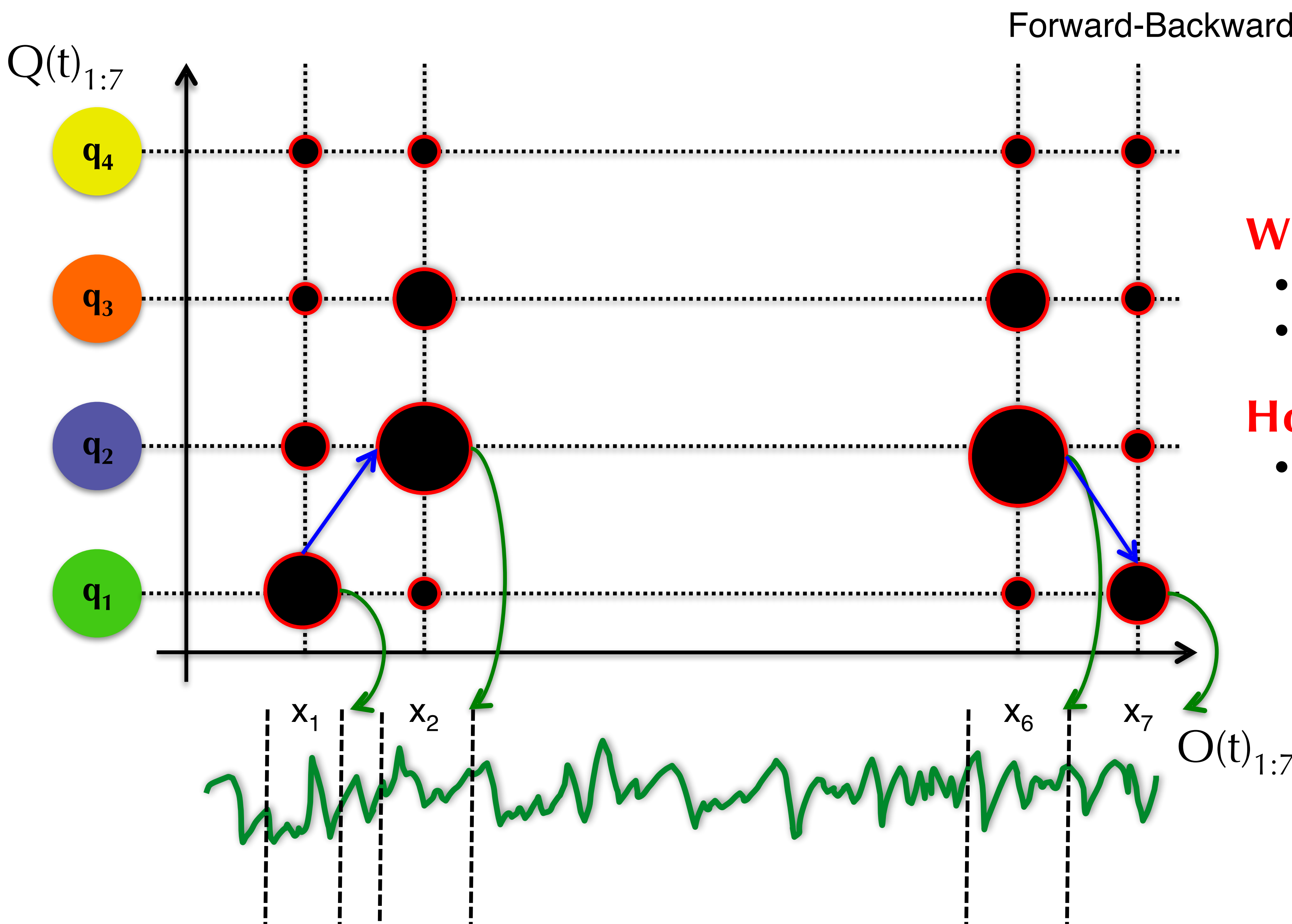
We know:

- the model M_3
- the sequence $O(t)_{1:7}$

Which are:

- the $Q(t)_{1:7}$ that generated $O(t)_{1:7}$ and maximizes $P(Q|O, \lambda)$?

Example in Gesture Recognition
How to evaluate a sequence of observations

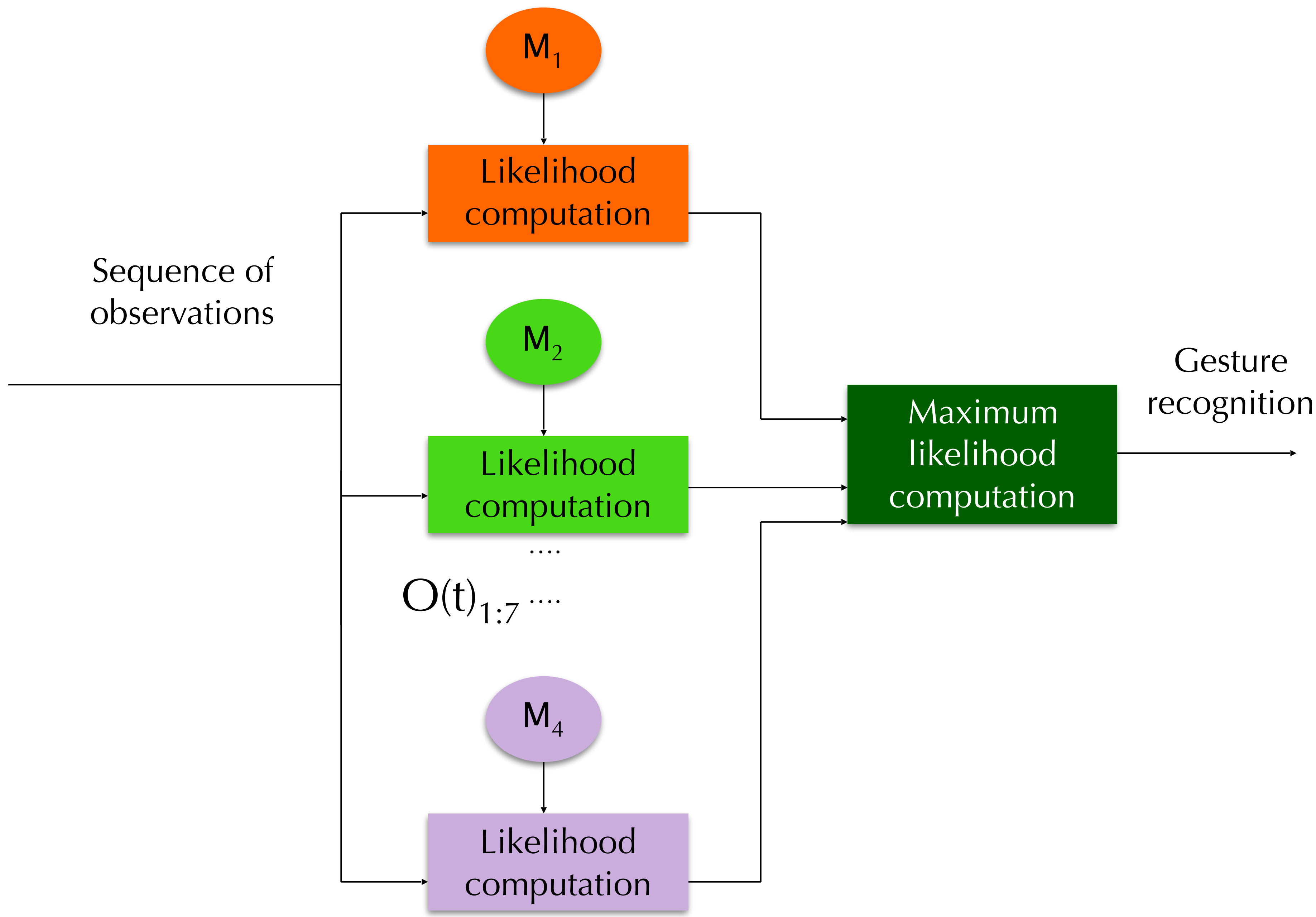


We know:
- the model M_3
- the sequence $O(t)_{1:7}$

How to:
- calculate $P(O(t)_{1:7} \mid M_3)$?

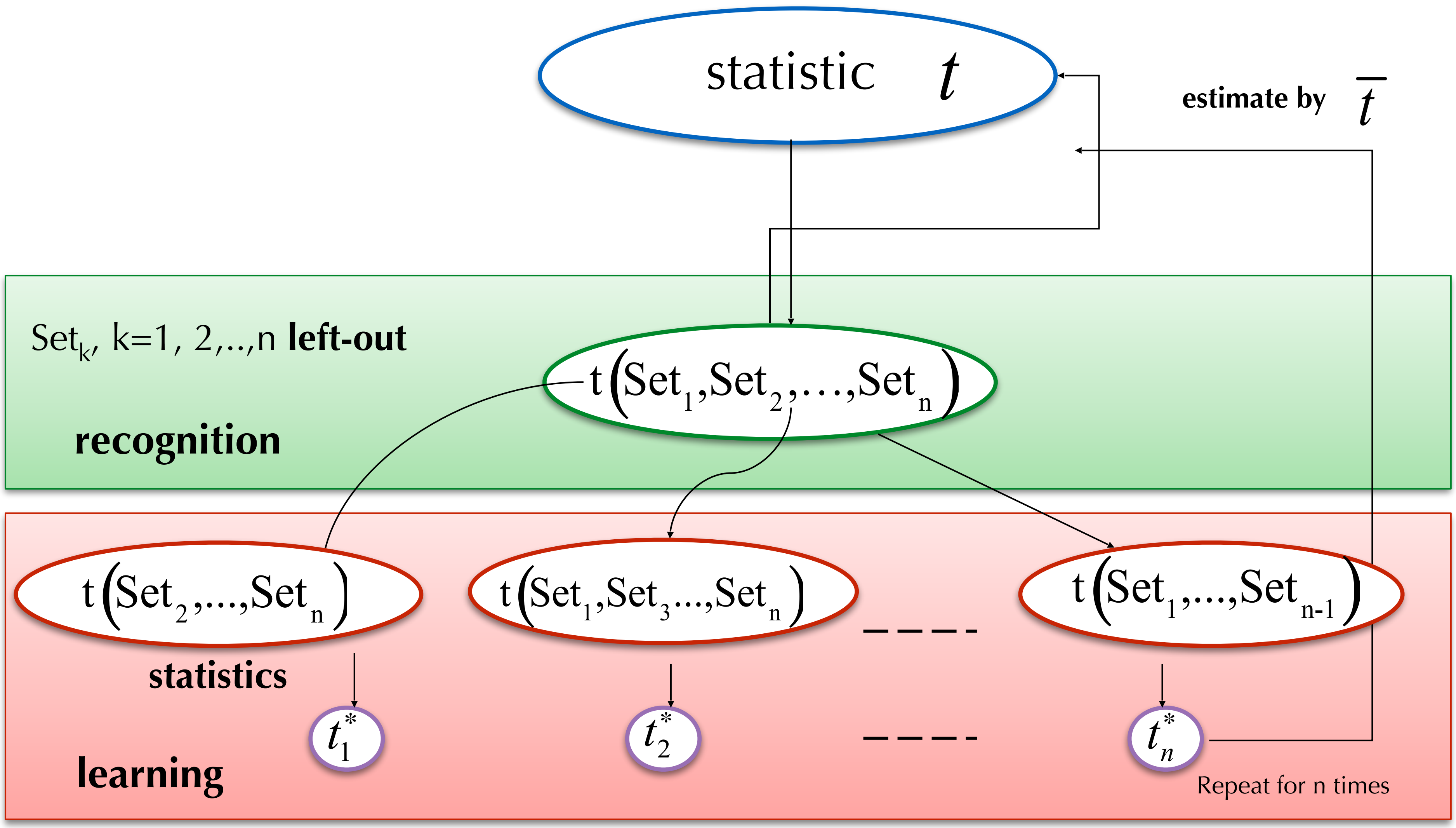
Example in Gesture Recognition

How to compute the recognize the gestures



Example in Gesture Recognition

How to evaluate the system



Example in Gesture Recognition
Statistics to be computed

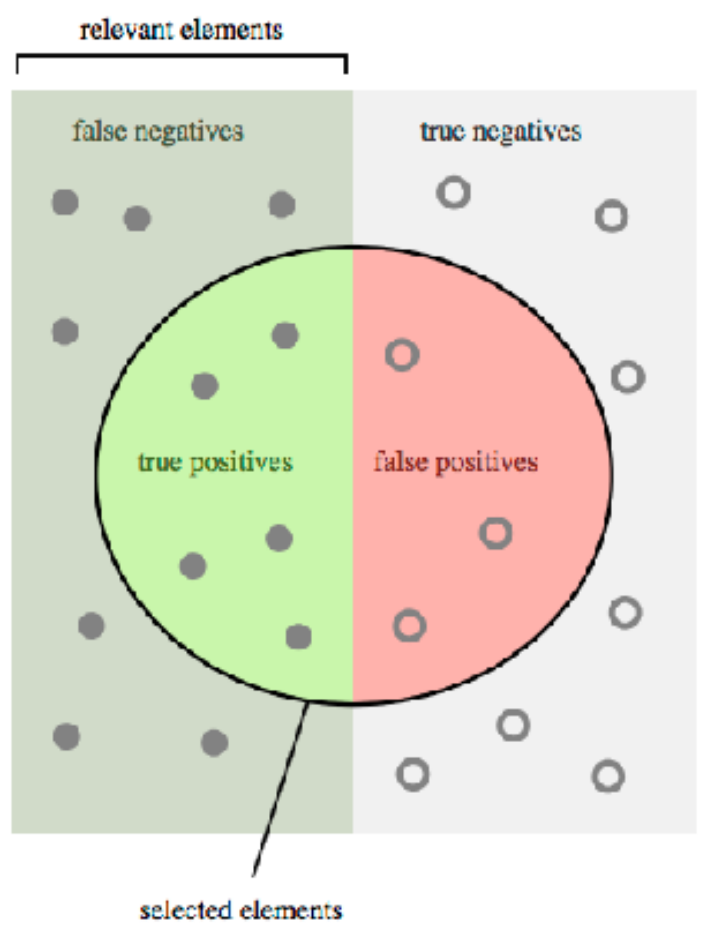


Table 4. Optical images: precision and recall based on jacknifing

		Number of maximum likelihoods				
		M_1^M	M_2^M	M_3^M	M_4^M	Recall (%)
Observations	G_1^M	70	-	20	-	77,8
	G_2^M	-	76	-	14	84,4
	G_3^M	8	-	82	-	91,1
	G_4^M	-	7	-	83	92,2
Precision (%)		89,7	91,6	80,4	85,6	

Table 5. Depth images: precision and recall based on jacknifing

		Number of maximum likelihoods				
		M_1^M	M_2^M	M_3^M	M_4^M	Recall (%)
Observations	G_1^M	82	-	8	-	91,1
	G_2^M	-	80	-	10	88,9
	G_3^M	17	-	73	-	81,1
	G_4^M	-	1	-	89	98,9
Precision (%)		82,8	98,8	90,1	89,9	