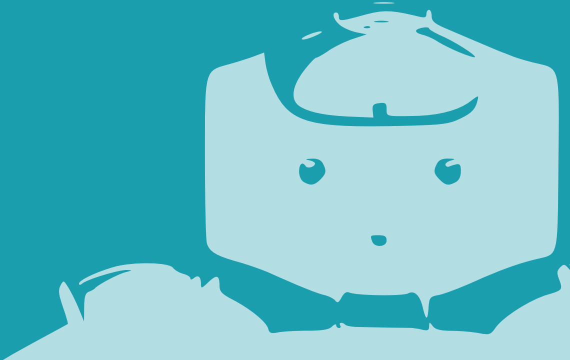
You can download the sources of this presentation here:
github.com/severin-lemaignan/lecture-hri-social-signal-processing

Human-Robot Interaction

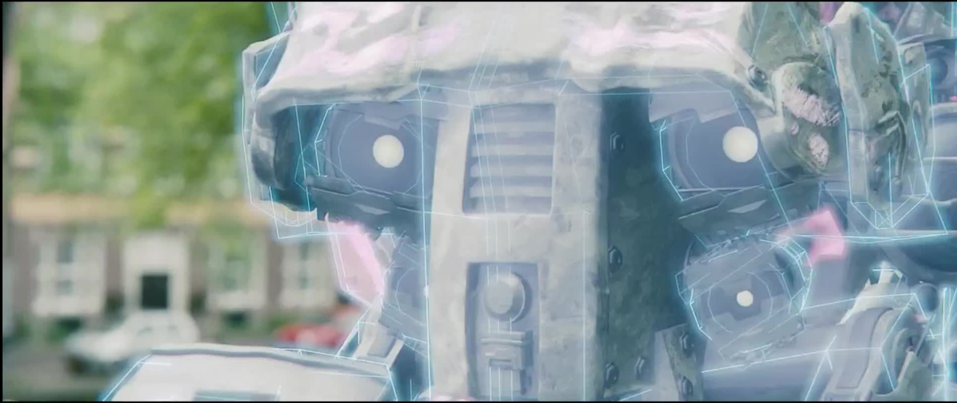
Social Signal Processing

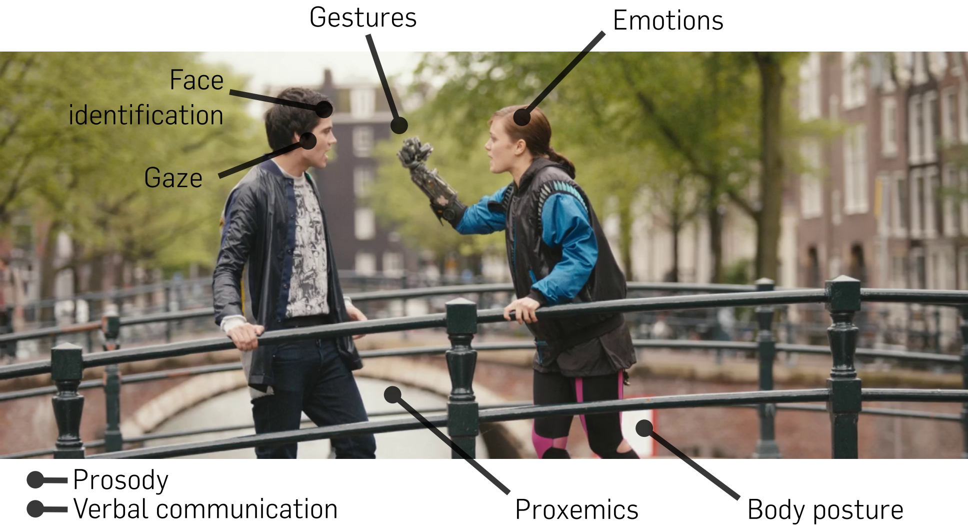


Séverin Lemaignan

Bristol Robotics Lab
University of the West of England

WHAT ARE SOCIAL SIGNALS?




Gestures
Emotions
Face identification
Gaze
Prosody
Verbal communication
Proxemics
Body posture

IN THIS LECTURE

1. What/why social signal processing?
2. Features
3. Example of facial action units
4. Principal Component Analysis and application to face recognition
5. From social signal to internal state inference

WHAT ARE SOCIAL SIGNALS?

- Social signals are ***observable*** behaviours that people display during social interactions

[Burgoon, Magnenat-Thalmann, Pantic, Vinciarelli, **Social Signal Processing**, 2017]

WHAT ARE SOCIAL SIGNALS?

- Social signals are ***observable*** behaviours that people display during social interactions
- Social signals from an individual ***produces changes*** in others (like creating a belief about the person, generating an appropriate social response, perform an actions)

[Burgoon, Magnenat-Thalmann, Pantic, Vinciarelli, **Social Signal Processing**, 2017]

WHAT ARE SOCIAL SIGNALS?

- Social signals are ***observable*** behaviours that people display during social interactions
- Social signals from an individual ***produces changes*** in others (like creating a belief about the person, generating an appropriate social response, perform an actions)
- the changes are not random, they follow ***principles and laws*** (in particular, ***social norms***)

[Burgoon, Magnenat-Thalmann, Pantic, Vinciarelli, **Social Signal Processing**, 2017]

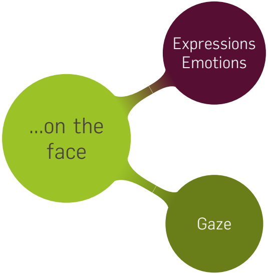
WHAT ARE SOCIAL SIGNALS?

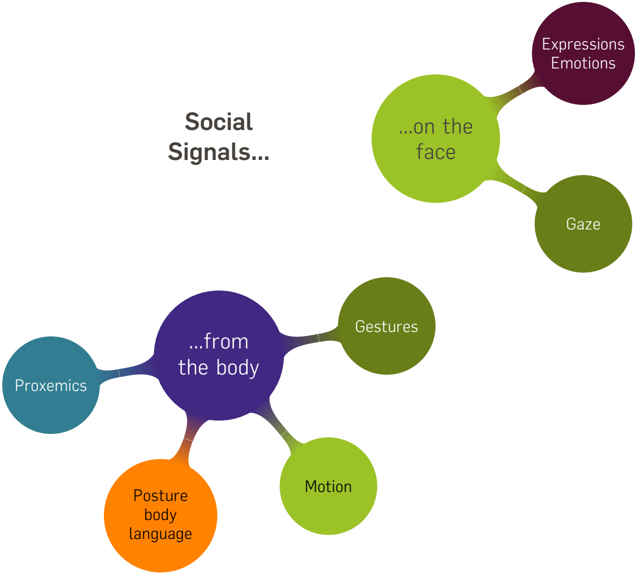
- Social signals are ***observable*** behaviours that people display during social interactions
- Social signals from an individual ***produces changes*** in others (like creating a belief about the person, generating an appropriate social response, perform an actions)
- the changes are not random, they follow ***principles and laws*** (in particular, ***social norms***)

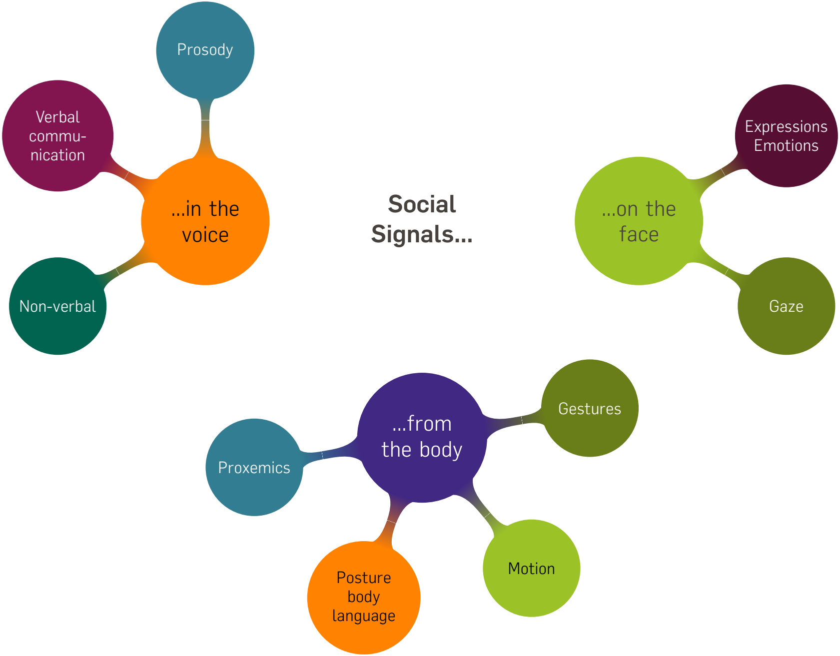
- Social signals are also a “window” into someone else ***internal state*** (physiological state, mental state, emotional state): **essential for the robot to generate appropriate behaviour!**

[Burgoon, Magnenat-Thalmann, Pantic, Vinciarelli, **Social Signal Processing**, 2017]

Social Signals...




Social Signals…
…on the face
Expressions Emotions
Gaze
…from the body
Gestures
Proxemics
Posture body language
Motion


Social Signals…
…in the voice
Prosody
Verbal commu-nication
Non-verbal
…on the face
Expressions Emotions
Gaze
…from the body
Gestures
Proxemics
Posture body language
Motion

WHY?

The ability to recognize human social signals and social behaviours like turn taking, politeness, and disagreement is essential when building social robots, human-robot interaction, or interactive systems

WHY?

The ability to recognize human social signals and social behaviours like turn taking, politeness, and disagreement is essential when building social robots, human-robot interaction, or interactive systems

3 main problems

- *Modeling*: identification of the principles and laws
- *Analysis*: automatic detection and interpretation
- *Synthesis*: automatic generation of artificial social signals

FEATURES

In order to understand "what's going on?" (usually reduced to a **classification** task), we first need to build a **representation**.

FEATURES

In order to understand "what's going on?" (usually reduced to a **classification** task), we first need to build a **representation**.

Building a representation often requires raw data **pre-processing** to extract **features**.

Features can be manually designed (*feature engineering*) or can be learnt (*supervised* or *unsupervised machine learning*)

FEATURES

In order to understand "what's going on?" (usually reduced to a **classification** task), we first need to build a **representation**.

Building a representation often requires raw data **pre-processing** to extract **features**.

Features can be manually designed (***feature engineering***) or can be learnt (***supervised*** or ***unsupervised machine learning***)

Feature engineering or supervised learning relies on pre-existing **models** (skeletons, language, eye, etc.), with ***unsupervised feature learning*** rely on the algorithm finding ***patterns*** in the data (that might or might not relate to recognisable models).

FEATURES

In order to understand “what's going on?” (usually reduced to a **classification** task), we first need to build a **representation**.

Building a representation often requires raw data **pre-processing** to extract **features**.

Features can be manually designed (***feature engineering***) or can be learnt (***supervised*** or ***unsupervised machine learning***)

Feature engineering or supervised learning relies on pre-existing **models** (skeletons, language, eye, etc.), with ***unsupervised feature learning*** rely on the algorithm finding ***patterns*** in the data (that might or might not relate to recognisable models).

Features can be more or less complex, from the age of a participant, to a complex social gesture.

FEATURES

What data features do you know of?

Take 5 minutes and list 3 ***features*** for each of the following sources of data:

- image of a face
- audio recording of a voice
- x,y,z coordinates of people in a crowd
- depth image of a person

FEATURES

- image of a face
    - skin colour
    - head pose, gaze direction
    - facial landmarks (eg contours)
    - action-units (more about that later!)
- audio recording of a voice
    - amplitude, frequencies
    - prosody
    - verbal content!
- x,y,z coordinates of people in a crowd
    - trajectories
    - proxemics
- depth image of a person
    - skeleton
    - gestures

FEATURES

- image of a face
  - skin colour
  - head pose, gaze direction
  - facial landmarks (eg contours) $\leftarrow$ **learnt**
  - action-units (more about that later!) $\leftarrow$ **learnt**
- audio recording of a voice
  - amplitude, frequencies
  - prosody $\leftarrow$ **learnt**
  - verbal content! $\leftarrow$ **learnt**
- x,y,z coordinates of people in a crowd
  - trajectories
  - proxemics
- depth image of a person
  - skeleton $\leftarrow$ **learnt**
  - gestures $\leftarrow$ **learnt**

EXAMPLE: RECOGNISING GENDER FROM SPEECH

Can we automatically recognise someone's gender from speech?

3,168 recorded voice samples, collected from male and female speakers.

- Examples from the database: male (US), female (US), male (Scotish)

The voice samples are pre-processed by acoustic analysis to extract 20 features (like mean frequency of the sample, spectral flatness, etc).

Source: data and more information

EXAMPLE: RECOGNISING GENDER FROM SPEECH

meanfreq	sd	median	Q25	Q75	IQR	skew	kurt	sp.ent	sfm	mode	centroid	meanfun	minfun	maxfun	meandom	mindom	maxdom	dfrange	modindx	label
0.05978	0.06424	0.03203	0.01507	0.09019	0.07512	12.8635	274.403	0.89337	0.49192	0	0.05978	0.08428	0.0157	0.27586	0.00781	0.00781	0.00781	0	0	male
0.06601	0.06731	0.04023	0.01941	0.09267	0.07325	22.4233	634.614	0.89219	0.51372	0	0.06601	0.10794	0.01583	0.25	0.00901	0.00781	0.05469	0.04688	0.05263	male
0.07732	0.08383	0.03672	0.0087	0.13191	0.12321	30.7572	1024.93	0.84639	0.47891	0	0.07732	0.09871	0.01566	0.27119	0.00799	0.00781	0.01563	0.00781	0.04651	male
0.192275	0.060818	0.21913	0.130952	0.242491	0.111539	1.891999	6.680009	0.915249	0.461751	0.244465	0.192275	0.114544	0.016615	0.210526	0.518682	0.03125	4.164063	4.132813	0.118491	male
0.203083	0.058876	0.232857	0.134286	0.247143	0.112857	2.858921	12.34421	0.852594	0.320577	0.246429	0.203083	0.108871	0.023426	0.15534	0.4375	0.21875	0.734375	0.515625	0.296296	male
0.166858	0.076829	0.202062	0.112096	0.228522	0.116426	1.971154	7.121826	0.937182	0.624363	0.216014	0.166858	0.095246	0.016563	0.142857	0.310547	0.15625	0.734375	0.578125	0.3	male
0.187391	0.059659	0.202846	0.125692	0.235809	0.110116	1.722606	6.693799	0.923242	0.464031	0.232549	0.187391	0.098694	0.027972	0.141593	0.324405	0.164063	0.59375	0.429688	0.324545	male
0.194088	0.061379	0.216466	0.127631	0.246827	0.119197	1.490315	4.321145	0.88923	0.364435	0.249639	0.194088	0.10925	0.036782	0.231884	0.495793	0.117188	2.164063	2.046875	0.192748	male
0.185908	0.062359	0.198432	0.133178	0.242034	0.108856	1.396273	5.493992	0.934298	0.521624	0.260127	0.185908	0.11307	0.020434	0.195122	0.696514	0.140625	5.414063	5.273438	0.165689	male
0.178028	0.070548	0.19	0.127436	0.242821	0.115385	2.149795	9.1742	0.945836	0.637708	0.251795	0.178028	0.116113	0.020101	0.275862	0.983854	0.03125	5.109375	5.078125	0.225199	male
0.187952	0.065855	0.203059	0.132068	0.245099	0.113031	1.480657	5.018319	0.934454	0.582426	0.243909	0.187952	0.111375	0.019704	0.181818	0.519737	0.0625	2.84375	2.78125	0.17603	male
0.208232	0.033483	0.214075	0.186861	0.231538	0.044678	2.569042	10.79804	0.86938	0.16029	0.226299	0.208232	0.186654	0.023121	0.258065	0.79974	0.171875	3.242188	3.070313	0.229271	female
0.199387	0.03544	0.196046	0.180719	0.226471	0.045752	2.342196	9.294968	0.860738	0.210884	0.18415	0.199387	0.159956	0.027119	0.271186	0.898438	0.007813	5.976563	5.96875	0.189915	female
0.195679	0.031613	0.193891	0.181785	0.21166	0.029874	3.189926	15.15865	0.850763	0.201765	0.193501	0.195679	0.183362	0.017778	0.25	0.935397	0.1875	5.921875	5.734375	0.193079	female
0.195605	0.033526	0.197941	0.179728	0.210751	0.031022	3.079277	14.56234	0.861835	0.22135	0.197341	0.195605	0.182781	0.029685	0.266667	1.055226	0.015625	6.25	6.234375	0.196491	female
0.200325	0.033138	0.205588	0.179753	0.223236	0.043483	2.107451	7.372721	0.869482	0.179791	0.223418	0.200325	0.17806	0.037915	0.266667	1.13151	0.164063	5.609375	5.445313	0.202108	female
0.212681	0.042392	0.212152	0.180529	0.25138	0.070851	1.142382	3.271853	0.895565	0.198001	0.19654	0.212681	0.169677	0.017837	0.266667	1.740885	0.148438	7	6.851563	0.35467	female
0.198039	0.030396	0.198105	0.183464	0.212974	0.02951	2.118279	7.139244	0.857263	0.177914	0.189412	0.198039	0.188897	0.025932	0.242424	0.508878	0.109375	1.507813	1.398438	0.324904	female
0.218552	0.037574	0.220555	0.200416	0.246274	0.045858	2.4775	11.06604	0.877562	0.188399	0.220555	0.218552	0.182308	0.020725	0.275862	0.474609	0.007813	1.492188	1.484375	0.199624	female
0.198203	0.031488	0.195094	0.182032	0.218269	0.036238	2.643353	12.04094	0.862682	0.174607	0.183085	0.198203	0.186066	0.07619	0.238806	0.714154	0.171875	6.171875	6	0.113542	female
0.202647	0.031964	0.198973	0.184434	0.223804	0.03937	2.44728	10.35293	0.864479	0.165662	0.185904	0.202647	0.184609	0.021769	0.25	1.107759	0.070313	6.140625	6.070313	0.19701	female
0.217759	0.031261	0.223285	0.1991	0.237762	0.038662	2.038032	6.67446	0.861819	0.15492	0.226691	0.217759	0.193159	0.017335	0.271186	1.109086	0.007813	5.914063	5.90625	0.177407	female
0.191456	0.030422	0.19173	0.172434	0.212874	0.04044	2.109024	7.296761	0.85787	0.175168	0.172023	0.191456	0.179518	0.028269	0.271186	0.642188	0.171875	3.429688	3.257813	0.174889	female

EXAMPLE: RECOGNISING GENDER FROM SPEECH

meanfreq	sd	median	Q25	Q75	IQR	skew	kurt	sp.ent	sfm	mode	centroid	meanfun	minfun	maxfun	meandom	mindom	maxdom	dfrange	modindx	label
0.05978	0.06424	0.03203	0.01507	0.09019	0.07512	12.8635	274.403	0.89337	0.49192	0	0.05978	0.08428	0.0157	0.27586	0.00781	0.00781	0.00781	0	0	male
0.06601	0.06731	0.04023	0.01941	0.09267	0.07325	22.4233	634.614	0.89219	0.51372	0	0.06601	0.10794	0.01583	0.25	0.00901	0.00781	0.05469	0.04688	0.05263	male
0.07732	0.08383	0.03672	0.0087	0.13191	0.12321	30.7572	1024.93	0.84639	0.47891	0	0.07732	0.09871	0.01566	0.27119	0.00799	0.00781	0.01563	0.00781	0.04651	male
0.192275	0.060818	0.21913	0.130952	0.242491	0.111539	1.891999	6.680009	0.915249	0.461751	0.244465	0.192275	0.114544	0.016615	0.210526	0.518682	0.03125	4.164063	4.132813	0.118491	male
0.203083	0.058876	0.232857	0.134286	0.247143	0.112857	2.858921	12.34421	0.852594	0.320577	0.246429	0.203083	0.108871	0.023426	0.15534	0.4375	0.21875	0.734375	0.515625	0.296296	male
0.166858	0.076829	0.202062	0.112096	0.228522	0.116426	1.971154	7.121826	0.937182	0.624363	0.216014	0.166858	0.095246	0.016563	0.142857	0.310547	0.15625	0.734375	0.578125	0.3	male
0.187391	0.059659	0.202846	0.125692	0.235809	0.110116	1.722606	6.693799	0.923242	0.464031	0.232549	0.187391	0.098694	0.027972	0.141593	0.324405	0.164063	0.59375	0.429688	0.324545	male
0.194088	0.061379	0.216466	0.127631	0.246827	0.119197	1.490315	4.321145	0.88923	0.364435	0.249639	0.194088	0.10925	0.036782	0.231884	0.495793	0.117188	2.164063	2.046875	0.192748	male
0.185908	0.062359	0.198432	0.133178	0.242034	0.108856	1.396273	5.493992	0.934298	0.521624	0.260127	0.185908	0.11307	0.020434	0.195122	0.696514	0.140625	5.414063	5.273438	0.165689	male
0.178028	0.070548	0.19	0.127436	0.242821	0.115385	2.149795	9.1742	0.945836	0.637708	0.251795	0.178028	0.116113	0.020101	0.275862	0.983854	0.03125	5.109375	5.078125	0.225199	male
0.187952	0.065855	0.203059	0.132068	0.245099	0.113031	1.480657	5.018319	0.934454	0.582426	0.243909	0.187952	0.111375	0.019704	0.181818	0.519737	0.0625	2.84375	2.78125	0.17603	male
0.208232	0.033483	0.214075	0.186861	0.231538	0.044678	2.569042	10.79804	0.86938	0.16029	0.226299	0.208232	0.186654	0.023121	0.258065	0.79974	0.171875	3.242188	3.070313	0.229271	female
0.199387	0.03544	0.196046	0.180719	0.226471	0.045752	2.342196	9.294968	0.860738	0.210884	0.18415	0.199387	0.159956	0.027119	0.271186	0.898438	0.007813	5.976563	5.96875	0.189915	female
0.195679	0.031613	0.193891	0.181785	0.21166	0.029874	3.189926	15.15865	0.850763	0.201765	0.193501	0.195679	0.183362	0.017778	0.25	0.935397	0.1875	5.921875	5.734375	0.193079	female
0.195605	0.033526	0.197941	0.179728	0.210751	0.031022	3.079277	14.56234	0.861835	0.22135	0.197341	0.195605	0.182781	0.029685	0.266667	1.055226	0.015625	6.25	6.234375	0.196491	female
0.200325	0.033138	0.205588	0.179753	0.223236	0.043483	2.107451	7.372721	0.869482	0.179791	0.223418	0.200325	0.17806	0.037915	0.266667	1.13151	0.164063	5.609375	5.445313	0.202108	female
0.212681	0.042392	0.212152	0.180529	0.25138	0.070851	1.142382	3.271853	0.895565	0.198001	0.19654	0.212681	0.169677	0.017837	0.266667	1.740885	0.148438	7	6.851563	0.35467	female
0.198039	0.030396	0.198105	0.183464	0.212974	0.02951	2.118279	7.139244	0.857263	0.177914	0.189412	0.198039	0.188897	0.025932	0.242424	0.508878	0.109375	1.507813	1.398438	0.324904	female
0.218552	0.037574	0.220555	0.200416	0.246274	0.045858	2.4775	11.06604	0.877562	0.188399	0.220555	0.218552	0.182308	0.020725	0.275862	0.474609	0.007813	1.492188	1.484375	0.199624	female
0.198203	0.031488	0.195094	0.182032	0.218269	0.036238	2.643353	12.04094	0.862682	0.174607	0.183085	0.198203	0.186066	0.07619	0.238806	0.714154	0.171875	6.171875	6	0.113542	female
0.202647	0.031964	0.198973	0.184434	0.223804	0.03937	2.44728	10.35293	0.864479	0.165662	0.185904	0.202647	0.184609	0.021769	0.25	1.107759	0.070313	6.140625	6.070313	0.19701	female
0.217759	0.031261	0.223285	0.1991	0.237762	0.038662	2.038032	6.67446	0.861819	0.15492	0.226691	0.217759	0.193159	0.017335	0.271186	1.109086	0.007813	5.914063	5.90625	0.177407	female
0.191456	0.030422	0.19173	0.172434	0.212874	0.04044	2.109024	7.296761	0.85787	0.175168	0.172023	0.191456	0.179518	0.028269	0.271186	0.642188	0.171875	3.429688	3.257813	0.174889	female

- kNN (k = 7): 97.8% classified correctly.
- SVM: 97.5% classified correctly.

→ Recognising gender from speech is easy and robust; many classification algorithms can deal with this problem.

(more on classification algorithms next week)

EXAMPLE: FACIAL ACTION UNITS

The **Facial Action Coding System** (FACS) aim is to **taxonomize human facial movements** by their appearance on the face.

Facial movements are encoded as **action units** → useful features to decode facial expressions

AU	Description	Example image
1	Inner Brow Raiser	
2	Outer Brow Raiser	
4	Brow Lowerer	
5	Upper Lid Raiser	

AU	Description	Example image
6	Cheek Raiser	
7	Lid Tightener	
9	Nose Wrinkler	
10	Upper Lip Raiser	
12	Lip Corner Puller	

[Ekman, **The Facial Action Coding System**, 1977]

EXAMPLE: FACIAL ACTION UNITS

AU	Description	Example image
15	Lip Corner Depressor	
17	Chin Raiser	
20	Lip stretcher	
23	Lip Tightener	

AU	Description	Example image
14	Dimpler	
25	Lips part	
26	Jaw Drop	
45	Blink	

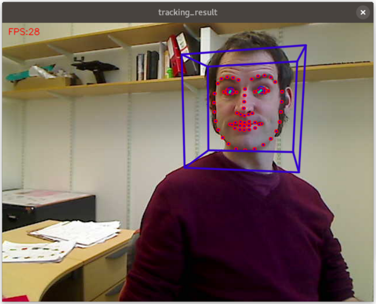
Wikipedia has a large list of action units.

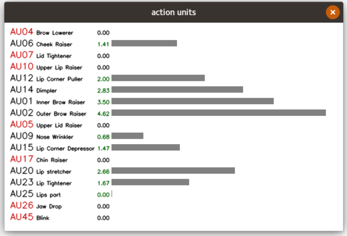
[Ekman, **The Facial Action Coding System**, 1977]

OPENFACE ACTION UNITS

OpenFace is an open-source library that recognises 17 action units (amongst many other things).

github.com/TadasBaltrusaitis/OpenFace

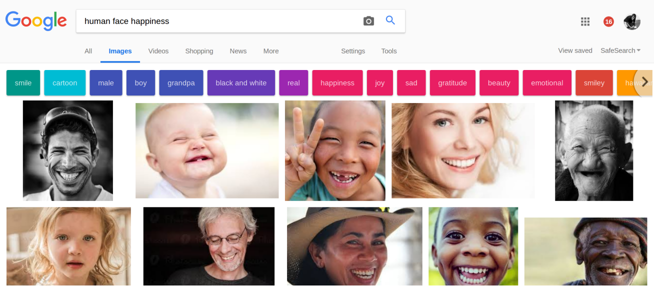




(not to be confused with this other CMU OpenFace)

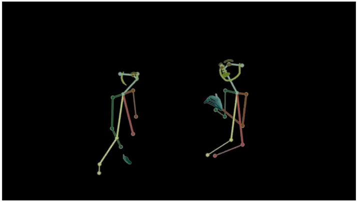
NEXT WEEK

Let’s build an emotion classifier from scratch!



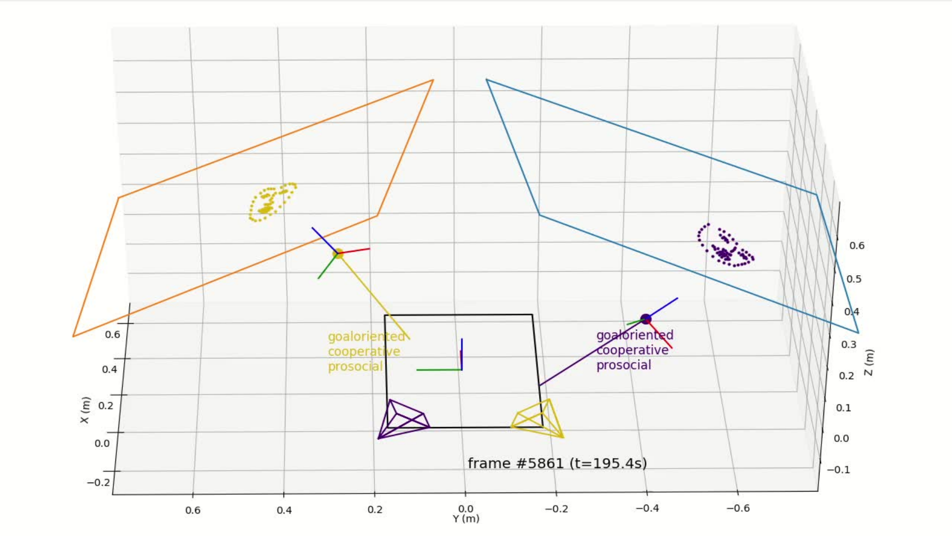
MULTI-MODAL EXAMPLE

In practice, our data is often multi-modal, and social signal processing usually requires multi-modal processing.

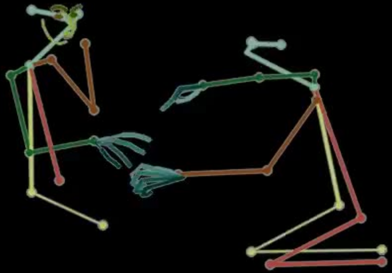


PInSoro dataset: a large dataset of multi-modal child-child (and child-robot interaction)

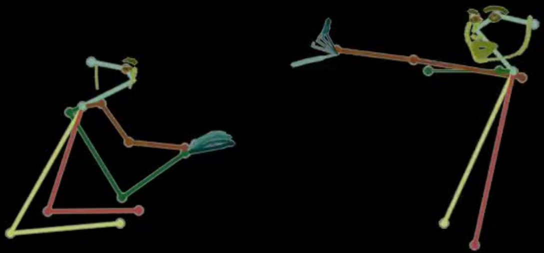
[Lemaignan et al. **The PInSoRo dataset: Supporting the data-driven study of child-child and child-robot [...]** PLOS One 2018]



goaloriented
cooperative
prosocial
goaloriented
cooperative
prosocial
frame #5861 (t=195.4s)
X (m)
Y (m)
Z (m)
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.1
0.1
0.3
0.5




Focused!
ILLUMINATED POLITICAL
& PHYSICAL GLOBE




open me

MULTI-MODAL REPRESENTATIONS

Building the right representation is hard, especially for multi-modal social signals (essentially, an open research question).

MULTI-MODAL REPRESENTATIONS

Building the right representation is hard, especially for multi-modal social signals (essentially, an open research question).

What social signals would you rely on for a robot to recognise that the little girl is bored?

MULTI-MODAL REPRESENTATIONS

Building the right representation is hard, especially for multi-modal social signals (essentially, an open research question).

What social signals would you rely on for a robot to recognise that the little girl is bored?

- gaze
- facial expression
- speech (or lack thereof!)
- (repetitive) gesture
- body language: bent over the table

MULTI-MODAL SOCIAL PROCESSING

Combining several modalities makes social signal processing more robust.

Example: while we usually recognise emotions from face images, adding voice is very useful:

Source: Berlin Database of Emotional Speech

„Sie haben es gerade hochgetragen und jetzt gehen sie wieder runter“ (They just carried it upstairs and now they are going down again).

Which emotion do you recognise?

Anger – Boredom – Disgust – Anxiety/Fear – Happiness – Sadness – Neutral

MULTI-MODAL SOCIAL PROCESSING

Combining several modalities makes social signal processing more robust.

Example: while we usually recognise emotions from face images, adding voice is very useful:

Source: Berlin Database of Emotional Speech

Prosody is an important *non-verbal* social signal.

In this dataset, humans were able to recognise emotions from prosody with more than 80% accuracy (***caveat: actors in a recording studio, not natural prosody!***)

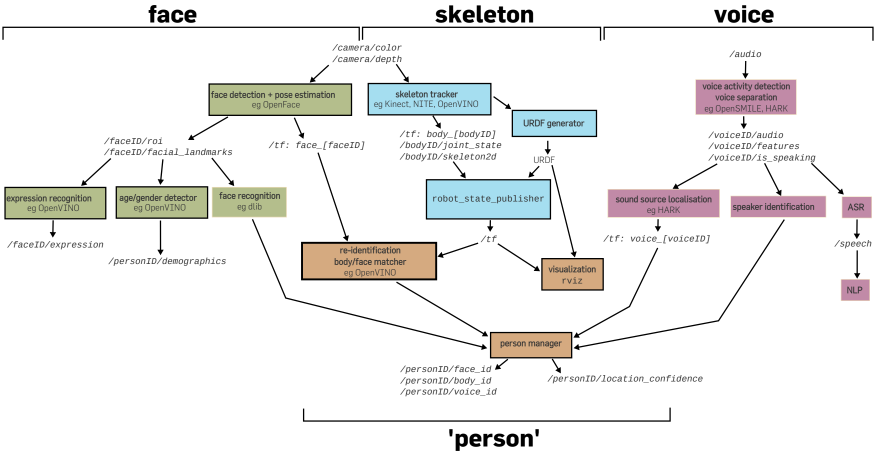
SOCIAL SIGNAL PROCESSING PIPELINE

Due to the complexity of real-world multi-modal signal, need to process the raw signal via a set of specialist functions, for instance:

- People detection
- Face recognition
- Gesture recognition
- Gaze detection
- Facial expression reading (wink, blink, talking, ...)
- Detection of social signals from verbal communication
- Emotion recognition
  - from faces
  - movement
  - speech ...
- ...

These tasks are typicaly executed **in parallel** and in a **pipeline**.

EXAMPLE: THE ROS4HRI PIPELINE



[*Source: Mohamed, Lemaignan,* **ROS for Human-Robot Interaction**, *2021*]

PRINCIPAL COMPONENT ANALYSIS: EXAMPLE OF FACE RECOGNITION

?

PRINCIPAL COMPONENT ANALYSIS

Principal Component Analysis (PCA) is a technique to ***find the source of variance in a dataset***.

PCA is a fundamental tool in data science, and a ***basic building block for social signal processing and analysis***.

(it is one of the simplest and most effective ***dimensionality reduction technique***, but other exist! eg (deep) autoencoder)

[UK food consumption in 1997, DEFRA, *Source: Mark Richardson, Principal Component Analysis*]

PRINCIPAL COMPONENT ANALYSIS

	England	N Ireland	Scotland	Wales
Alcoholic drinks	375	135	458	475
Beverages	57	47	53	73
Carcase meat	245	267	242	227
Cereals	1472	1494	1462	1582
Cheese	105	66	103	103
Confectionery	54	41	62	64
Fats and oils	193	209	184	235
Fish	147	93	122	160
Fresh fruit	1102	674	957	1137
Fresh potatoes	720	1033	566	874
Fresh Veg	253	143	171	265
Other meat	685	586	750	803
Other Veg	488	355	418	570
Processed potatoes	198	187	220	203
Processed Veg	360	334	337	365
Soft drinks	1374	1506	1572	1256
Sugars	156	139	147	175

Question: what food preference distinguishes best the four nations?

[UK food consumption in 1997, DEFRA, *Source: Mark Richardson, Principal Component Analysis*]

PRINCIPAL COMPONENT ANALYSIS

4 classes of observations

17 features

	England	N Ireland	Scotland	Wales
Alcoholic drinks	375	135	458	475
Beverages	57	47	53	73
Carcase meat	245	267	242	227
Cereals	1472	1494	1462	1582
Cheese	105	66	103	103
Confectionery	54	41	62	64
Fats and oils	193	209	184	235
Fish	147	93	122	160
Fresh fruit	1102	674	957	1137
Fresh potatoes	720	1033	566	874
Fresh Veg	253	143	171	265
Other meat	685	586	750	803
Other Veg	488	355	418	570
Processed potatoes	198	187	220	203
Processed Veg	360	334	337	365
Soft drinks	1374	1506	1572	1256
Sugars	156	139	147	175

Same question: what linear combination of features maximize the variance in the dataset? $\Rightarrow$ PCA!

[UK food consumption in 1997, DEFRA, *Source: Mark Richardson, Principal Component Analysis*]

PRINCIPAL COMPONENT ANALYSIS

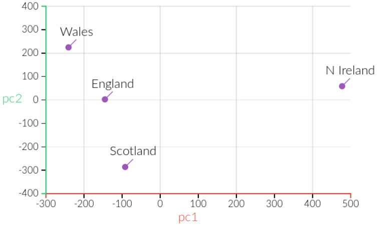
	England	N Ireland	Scotland	Wales
Alcoholic drinks	375	135	458	475
Beverages	57	47	53	73
Carcase meat	245	267	242	227
Cereals	1472	1494	1462	1582
Cheese	105	66	103	103
Confectionery	54	41	62	64
Fats and oils	193	209	184	235
Fish	147	93	122	160
Fresh fruit	1102	674	957	1137
Fresh potatoes	720	1033	566	874
Fresh Veg	253	143	171	265
Other meat	685	586	750	803
Other Veg	488	355	418	570
Processed potatoes	198	187	220	203
Processed Veg	360	334	337	365
Soft drinks	1374	1506	1572	1256
Sugars	156	139	147	175

The *first principal component* `pc1` is the best possible linear combination. Can you guess what it is?

[UK food consumption in 1997, DEFRA, *Source: Mark Richardson, Principal Component Analysis*]

PRINCIPAL COMPONENT ANALYSIS

	England	N Ireland	Scotland	Wales
Alcoholic drinks	375	135	458	475
Beverages	57	47	53	73
Carcase meat	245	267	242	227
Cereals	1472	1494	1462	1582
Cheese	105	66	103	103
Confectionery	54	41	62	64
Fats and oils	193	209	184	235
Fish	147	93	122	160
Fresh fruit	1102	674	957	1137
Fresh potatoes	720	1033	566	874
Fresh Veg	253	143	171	265
Other meat	685	586	750	803
Other Veg	488	355	418	570
Processed potatoes	198	187	220	203
Processed Veg	360	334	337	365
Soft drinks	1374	1506	1572	1256
Sugars	156	139	147	175

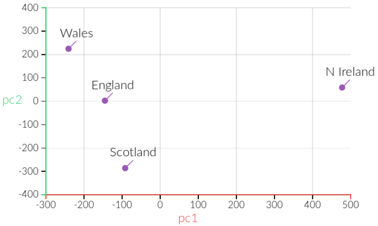


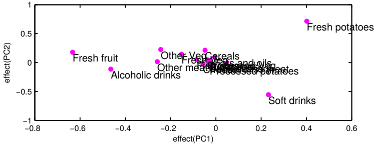
We can add more principal components to better ***explain the dataset variance***. From a classification perspective, 2 components seem sufficient to separate our 4 nations: We can **reduce** our 17 dimensions to only 2

[UK food consumption in 1997, DEFRA, *Source: Mark Richardson, Principal Component Analysis*]

PRINCIPAL COMPONENT ANALYSIS

	England	N Ireland	Scotland	Wales
Alcoholic drinks	375	135	458	475
Beverages	57	47	53	73
Carcase meat	245	267	242	227
Cereals	1472	1494	1462	1582
Cheese	105	66	103	103
Confectionery	54	41	62	64
Fats and oils	193	209	184	235
Fish	147	93	122	160
Fresh fruit	1102	674	957	1137
Fresh potatoes	720	1033	566	874
Fresh Veg	253	143	171	265
Other meat	685	586	750	803
Other Veg	488	355	418	570
Processed potatoes	198	187	220	203
Processed Veg	360	334	337	365
Soft drinks	1374	1506	1572	1256
Sugars	156	139	147	175





$pc_1 \approx 0.4 \times \text{potatoes} + 0.2 \times \text{softs} - 0.4 \times \text{alcohol} - 0.6 \times \text{fruits}$

$pc_2 \approx 0.55 \times \text{potatoes} - 0.5 \times \text{softs}$

PCA ALGORITHM

Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be a vector with observations $\mathbf{x}_i \in \mathbb{R}^d$.

1. Compute the mean μ

$$\mu = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i$$

2. Compute the the Covariance Matrix S

$$S = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T$$

3. Compute the eigenvalues λ_i and eigenvectors $\mathbf{v}_i$ of S

$$S \cdot \mathbf{v}_i = \lambda_i \mathbf{v}_i \quad \text{with } i = 1, 2, \ldots, n$$

4. Order the eigenvectors descending by their eigenvalue. The k principal components are the eigenvectors corresponding to the k largest eigenvalues.

[*Source: Philipp Wagner, Face Recognition with Python* (code source available as well)]

PYTHON CODE

```
def pca(X):

    mu = X.mean(axis=0)
    X = X - mu
    C = np.dot(X.T,X)
    eigenvalues, eigenvectors = np.linalg.eigh(C)

    # sort eigenvectors descending by their eigenvalue
    idx = np.argsort(-eigenvalues)
    eigenvalues = eigenvalues[idx]
    eigenvectors = eigenvectors[:,idx]
    return eigenvalues, eigenvectors, mu

# D: eigenvalues, W: eigenvectors, mu: mean, X: dataset
D, W, mu = pca(X)
```

or:

```
from sklearn.decomposition import PCA
pca = PCA(n_components=2) # for instance, only keep 2 eigenvectors
pca.fit(X) # calculate the eigenvalues/eigenvectors of X
Y_pca = pca.transform(Y) # apply the dimensionality reduction
pca.explained_variance_ratio_ # variance explained by each eigenvector
```

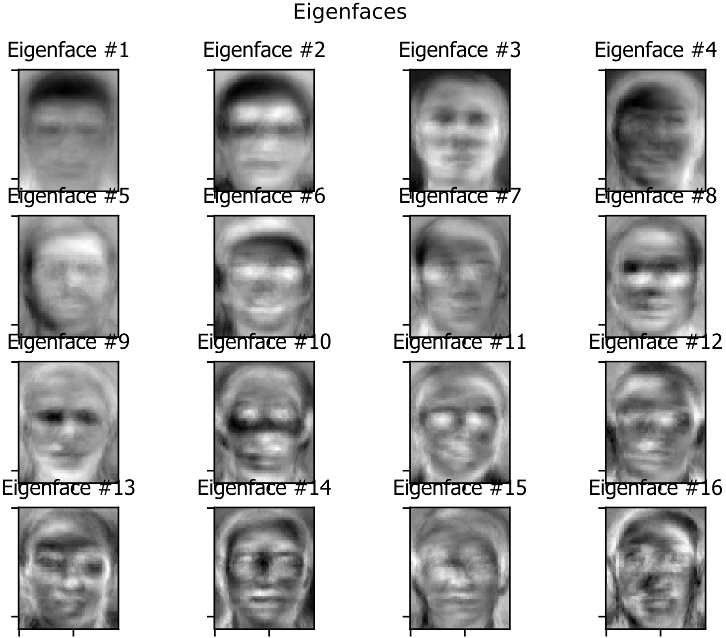
AT&T Face dataset

Applied to faces

EIGENFACES: PYTHON CODE

```
from pathlib import Path
from PIL import Image
import numpy as np
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA

# load the images
images = []
for img in Path("att_faces").glob("*/*.pgm"):
    face = Image.open(img)
    images.append(np.asarray(face).flatten())

# compute the first 16 eigenvectors, eg 'eigenfaces'
X = np.asarray(images) # 400 X 10304 image array
pca = PCA(n_components=16)
pca.fit(X)

# reconstruct and plot 16 'eigenfaces'
f, axarr = plt.subplots(4,4)
for i in range(16):
    image = pca.components_[i,:].reshape(112,92)
    axarr[i%4,i//4].imshow(image * 255)
plt.show()
```

AT&T Face dataset


Eigenfaces
Eigenface #1
Eigenface #2
Eigenface #3
Eigenface #4
Eigenface #5
Eigenface #6
Eigenface #7
Eigenface #8
Eigenface #9
Eigenface #10
Eigenface #11
Eigenface #12
Eigenface #13
Eigenface #14
Eigenface #15
Eigenface #16

PCA PROJECTION AND RECONSTRUCTION

The k principal components of an observed vector $\mathbf{x}$ are then given by:

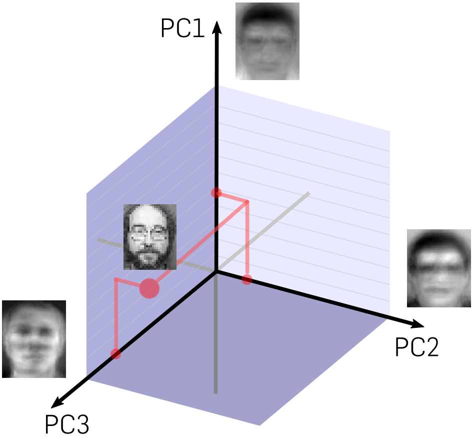
The image of a face!

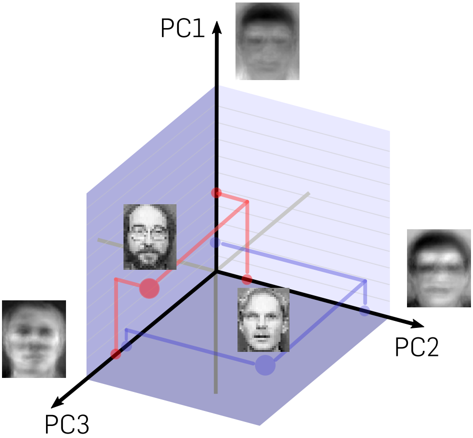
$$\mathbf{y} = [\mathbf{x}_{pc_1}, \ldots, \mathbf{x}_{pc_k}] = W^T(\mathbf{x} - \mu)$$

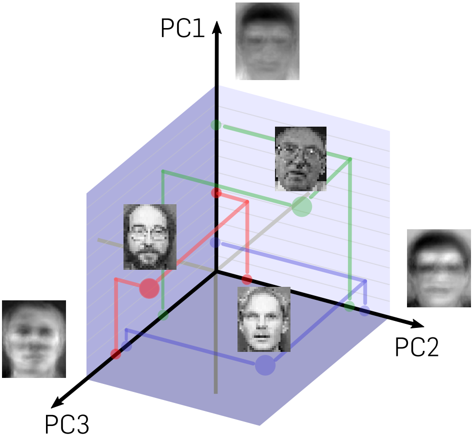
The PCA basis

where $W = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k)$, with $\mathbf{v}_k$ the eigenvectors.

$\mathbf{y}$ is the **projection** of $\mathbf{x}$ onto W.


PC1
PC2
PC3


PC1
PC2
PC3


PC1
PC2
PC3

PCA PROJECTION AND RECONSTRUCTION

The k principal components of an observed vector $\mathbf{x}$ are then given by:

The image of a face!

$$\mathbf{y} = [\mathbf{x}_{pc_1}, \ldots, \mathbf{x}_{pc_k}] = W^T(\mathbf{x} - \mu)$$

The PCA basis

where $W = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k)$, with $\mathbf{v}_k$ the eigenvectors.

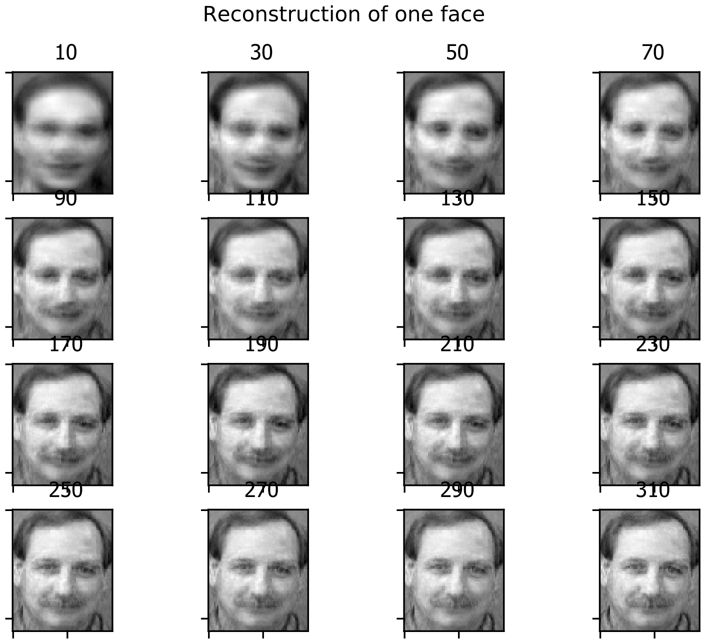
$\mathbf{y}$ is the **projection** of $\mathbf{x}$ onto W.

The reconstruction from the PCA basis is given by:

$$\mathbf{x} = W \cdot \mathbf{y} + \mu$$

PYTHON CODE: FACE RECONSTRUCTION

```
W = pca.components_
mu = pca.mean_

def project(W, x, mu=None):
    if mu is None:
        return np.dot(W,x)
    return np.dot(W,x - mu)

def reconstruct(W, y, mu=None):
    if mu is None:
        return np.dot(W.T,y)
    return np.dot(W.T,y) + mu

images = []; f, axarr = plt.subplots(4,4)
for nb_evs in range(10, 310, 20):
    p = project(W[0:nb_evs,:], X[0], mu) # you can also use pca.transform
    r = reconstruct(W[0:nb_evs,:], p, mu) # ...and pca.inverse_transform
    images.append(r.reshape(112,92))

for i in range(len(images)):
    axarr[i//4,i%4].imshow(images[i]*255)
plt.show()
```


Reconstruction of one face
10
30
50
70
90
110
130
150
170
190
210
230
250
270
290
310

WHY IS IT USEFUL?

Original images: $\textit{dim}(\mathbf{x}) = 92 \times 112 = 10304$ pixels: large number of dimensions!

$\Rightarrow$ difficult to tell whether 2 images represent the same person (i.e. ***classify*** them).

WHY IS IT USEFUL?

Original images: $\mathit{dim}(\mathbf{x}) = 92 \times 112 = 10304$ pixels: large number of dimensions!

$\Rightarrow$ difficult to tell whether 2 images represent the same person (i.e. ***classify*** them).

With the PCA, we project our test image onto a PCA basis of k principal components: $\mathbf{y} = W^T(\mathbf{x} - \mu)$ with $W = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k)$.

$\mathit{dim}(\mathbf{y}) = k$ can typically be much smaller than $\mathit{dim}(\mathbf{x})$

WHY IS IT USEFUL?

Original images: $dim(\mathbf{x}) = 92 \times 112 = 10304$ pixels: large number of dimensions!

$\Rightarrow$ difficult to tell whether 2 images represent the same person (i.e. ***classify*** them).

With the PCA, we project our test image onto a PCA basis of k principal components: $\mathbf{y} = W^T(\mathbf{x} - \mu)$ with $W = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k)$.

$dim(\mathbf{y}) = k$ can typically be much smaller than $dim(\mathbf{x})$

We effectively "summarize" our image into a few key values, along the principal axes of variation of our dataset.

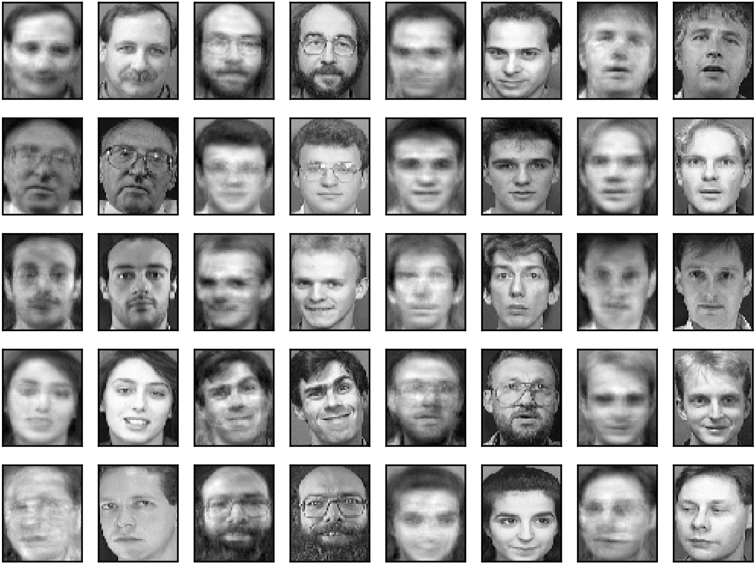
$\Rightarrow$ these values discriminate effectively amongst our images (maximise variance)

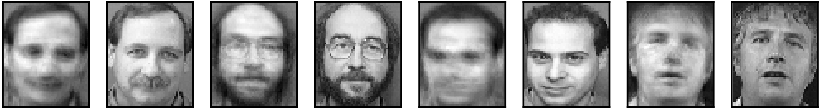
$\Rightarrow$ **Well suited for classification!**

Reconstruction with 1 Eigenvectors

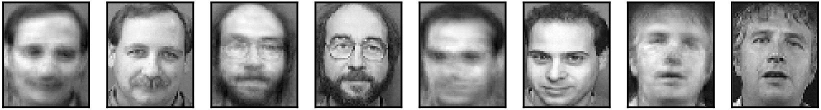
Reconstruction with 10 Eigenvectors

Reconstruction with 50 Eigenvectors



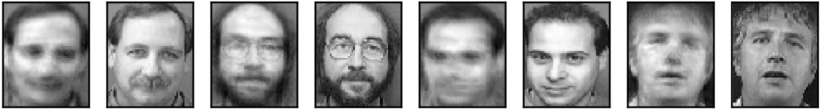


Remember: these faces are reconstructed from 50 values (to be compared to the 10304 values required for the original photos).



Remember: these faces are reconstructed from 50 values (to be compared to the 10304 values required for the original photos).

PCA is one example of a **dimensionality reduction** technique (i.e. a kind of lossy data compression).



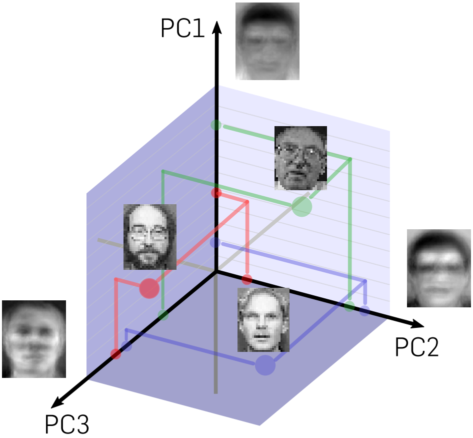
Remember: these faces are reconstructed from 50 values (to be compared to the 10304 values required for the original photos).

PCA is one example of a **dimensionality reduction** technique (i.e. a kind of lossy data compression).

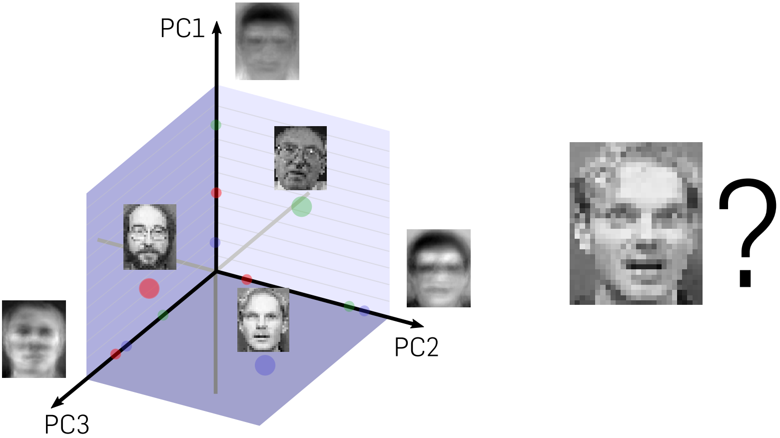
The resulting 50-dimensional vector is called an **embedding** of the face: a projection of a high dimensional representation (10304 pixels) into a much lower dimensional space (50 'eigenfaces').

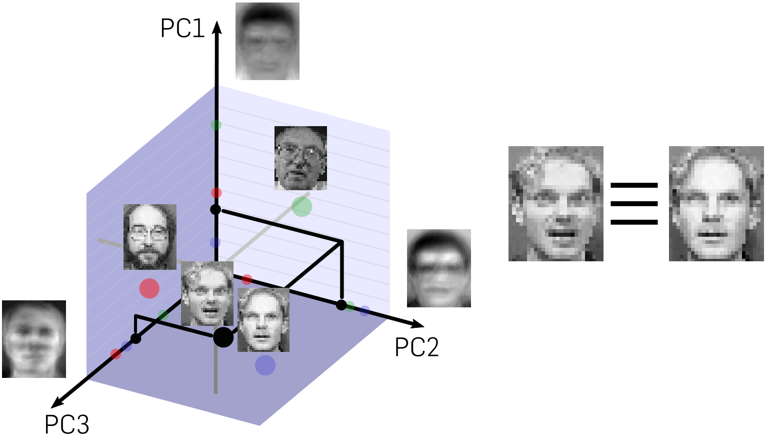
FACE RECOGNITION


?


PC1
PC2
PC3

?


PC1
PC2
PC3
?


PC1
PC2
PC3

RECOGNITION

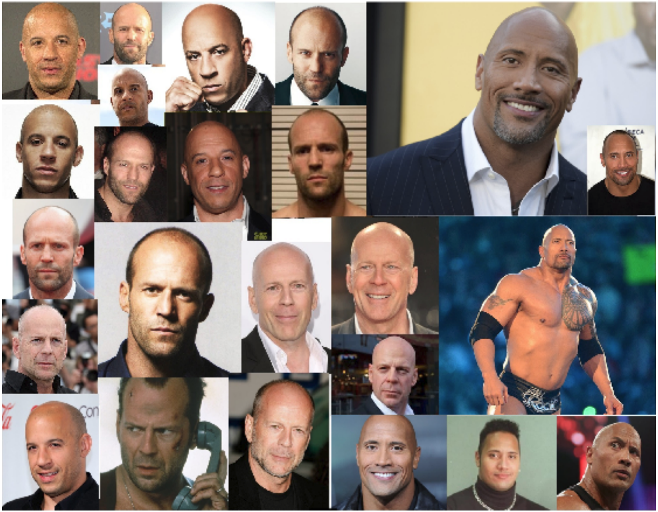
1. **learn a model** by (1) computing the PCA basis of the training set, (2) projecting each training face onto that basis
2. **project the test image** (eg, the face you want to recognise)
3. **find the 1-nearest neighbour**

PYTHON CODE

```
def dist(p, q):
    p = np.asarray(p).flatten()
    q = np.asarray(q).flatten()
    return np.sqrt(np.sum(
                np.power((p-q),2)
                ))


def learn_model(X):
    # compute PCA basis
    D, W, mu = pca(X, nb_evs=10)
    # compute projections
    projections = []
    for xi in X:
        yi = project(W, xi, mu)
        projections.append(yi)

    return W, projections
```

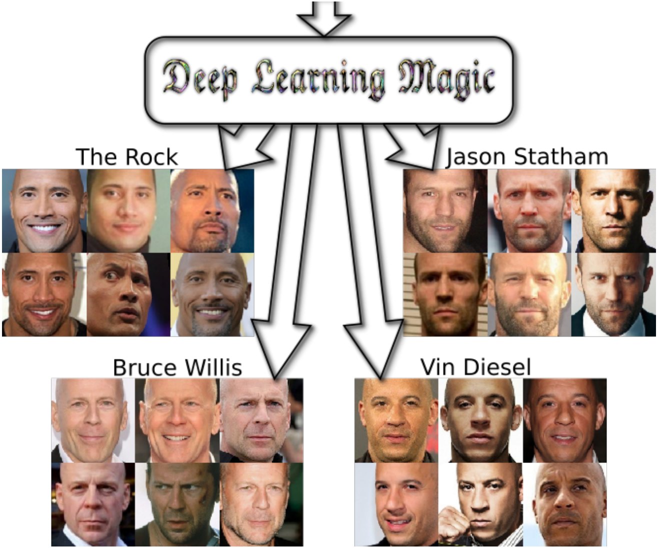
```
def predict(x, W, projections):
    minDist = np.finfo('float').max
    minClass = -1
    Q = project(W, x, mu)

    for i in range(len(projections)):
        dist = dist(projections[i], Q)
        if dist < minDist:
            minDist = dist
            faceClass = faceClasses[i]
    return faceClass


X, faceClasses = read_images()
W, projections = learn_model(X)
predict(test_image, W, projections)
```

STATE OF THE ART FACE RECOGNITION



[*Source: dlib* face recognition]

STATE OF THE ART FACE RECOGNITION



[*Source:* *dlib* face recognition]

FROM SOCIAL SIGNAL TO INTERNAL STATE

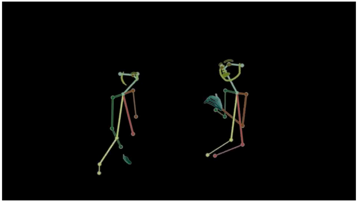
FROM SOCIAL SIGNAL TO INTERNAL STATE

For the robot to behave appropriately, it needs to assess the current state of the interaction, which typically requires estimating the ***internal state*** of the people: are they bored, excited, tired, curious,...?

FROM SOCIAL SIGNAL TO INTERNAL STATE

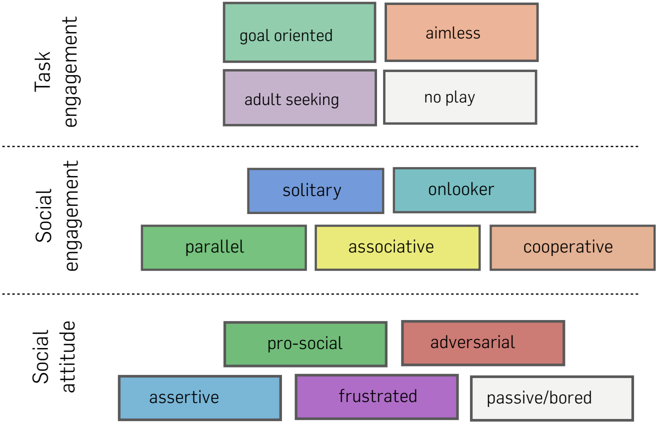
For the robot to behave appropriately, it needs to assess the current state of the interaction, which typically requires estimating the ***internal state*** of the people: are they bored, excited, tired, curious,...?

Question: can we infer the internal state of the children based on either image?



If a human can guess based on skeletons only, then there's good hope we can train a classifier for the robot to do the same.

13000+ ANNOTATIONS



[Lemaignan et al. **The PInSoRo dataset: Supporting the data-driven study of child-child...** PLOS One 2018]


Attitude: passive
Social engag.: onlooker
Task engag.: no play
Attitude: passive
Social engag.: solitary
Task engag.: goal oriented

- 20 clips extracted from the dataset, featuring notable social behaviours (boredom, aggression, cooperation, dominance, fun, excitement)

- 20 clips extracted from the dataset, featuring notable social behaviours (boredom, aggression, cooperation, dominance, fun, excitement)
- online study to ask people to rate the clips along 20 dimensions

[Bartlett, Edmunds, Belpaeme, Thill, Lemaignan, **What Can You See? Identifying Cues on Internal States...**, FrontiersIn 2019]

full-scene **OR** skeletons only

Page 1 of 4.

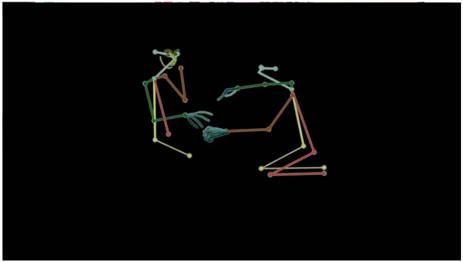
How much do you agree with the following statements?

The children were competing with one another.

Strongly Disagree | Disagree | Not Sure | Agree | Strongly Agree

The child on the left was sad.

Strongly Disagree | Disagree | Not Sure | Agree | Strongly Agree



Page 1 of 4.

How much do you agree with the following statements?

The children were competing with one another.

Strongly Disagree | Disagree | Not Sure | Agree | Strongly Agree

The child on the left was sad.

Strongly Disagree | Disagree | Not Sure | Agree | Strongly Agree

- 20 clips extracted from the dataset, featuring notable social behaviours (boredom, aggression, cooperation, dominance, fun, excitement)
- online study to ask people to rate the clips along 20 dimensions
- train a classifier to guess the social behaviour based on the ratings provided with the ***full video*** ⇒ resulting precision = 46%

- 20 clips extracted from the dataset, featuring notable social behaviours (boredom, aggression, cooperation, dominance, fun, excitement)
- online study to ask people to rate the clips along 20 dimensions
- train a classifier to guess the social behaviour based on the ratings provided with the ***full video*** $\Rightarrow$ resulting precision = 46%
- check how well a ***similar*** classifier performs when trained with the ratings provided for the ***skeleton-only*** videos $\Rightarrow$ precision = 42%

- 20 clips extracted from the dataset, featuring notable social behaviours (boredom, aggression, cooperation, dominance, fun, excitement)
- online study to ask people to rate the clips along 20 dimensions
- train a classifier to guess the social behaviour based on the ratings provided with the *full video* $\Rightarrow$ resulting precision = 46%
- check how well a ***similar*** classifier performs when trained with the ratings provided for the ***skeleton-only*** videos $\Rightarrow$ precision = 42%

The skeleton-only data seems to contain approx. the same amount of information on the internal state as the full-scene videos.

- 20 clips extracted from the dataset, featuring notable social behaviours (boredom, aggression, cooperation, dominance, fun, excitement)
- online study to ask people to rate the clips along 20 dimensions
- train a classifier to guess the social behaviour based on the ratings provided with the *full video* $\Rightarrow$ resulting precision = 46%
- check how well a ***similar*** classifier performs when trained with the ratings provided for the ***skeleton-only*** videos $\Rightarrow$ precision = 42%

The skeleton-only data seems to contain approx. the same amount of information on the internal state as the full-scene videos.

- 20 clips extracted from the dataset, featuring notable social behaviours (boredom, aggression, cooperation, dominance, fun, excitement)
- online study to ask people to rate the clips along 20 dimensions
- train a classifier to guess the social behaviour based on the ratings provided with the ***full video*** ⇒ resulting precision = 46%
- check how well a ***similar*** classifier performs when trained with the ratings provided for the ***skeleton-only*** videos ⇒ precision = 42%

The skeleton-only data seems to contain approx. the same amount of information on the internal state as the full-scene videos.

Humans can pick these complex social signal; robots not yet!

WE'VE BARELY SCRATCHED THE SURFACE!

Social signal processing is about extracting relevant information from the social environment.

Some techniques work relatively well:

- Face recognition, voice activity detection, gender classification, ...

Some work, but need improvement:

- Gaze detection, basic emotion recognition, speech recognition, ...

WE'VE BARELY SCRATCHED THE SURFACE!

But many open problems remain, eg:

- Complex real-word affect and emotion recognition (e.g. embarrassment, pride).
- Speech recognition for atypical speakers (children, elderly), multi-party interaction, ...
- most body language
- group dynamics

That's all for today, folks!

Questions:

severin.lemaignan@brl.ac.uk

Slides:

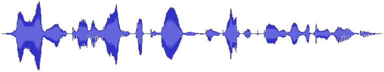
github.com/severin-lemaignan/lecture-social-signal-processing

CLASSIFICATION OF AUDITORY SIGNALS

Raw signals will in most cases require pre-processing to extract features.

The raw social signal (audio or video) requires pre-processing to extract between 10 and over a 1000 **features**.

- A raw signals contains too much data, and cannot be fed to the classifier immediately.



- Pre-processing extracts feature data which is relevant for the information which we are after (pitch, volume/energy, duration, formant frequencies, ...)
- These features then form the input for the classifier.

For more information see, for example, Liang et al. (2005) Feature analysis and extraction for audio automatic classification, Systems, Man and Cybernetics, 2005 IEEE International Conference on.

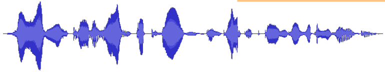
CLASSIFICATION OF AUDITORY SIGNALS

Raw signals will in most cases require pre-processing to extract features.

The raw social signal (audio or video) requires pre-processing to extract between 10 and over a 1000 **features**.

An exception to this are Convolutional Neural Networks, which can deal with unprocessed data

- A raw signals contains too much data, and cannot be fed to the classifier immediately.



- Pre-processing extracts feature data which is relevant for the information which we are after (pitch, volume/energy, duration, formant frequencies, ...)
- These features then form the input for the classifier.

For more information see, for example, Liang et al. (2005) Feature analysis and extraction for audio automatic classification, Systems, Man and Cybernetics, 2005 IEEE International Conference on.