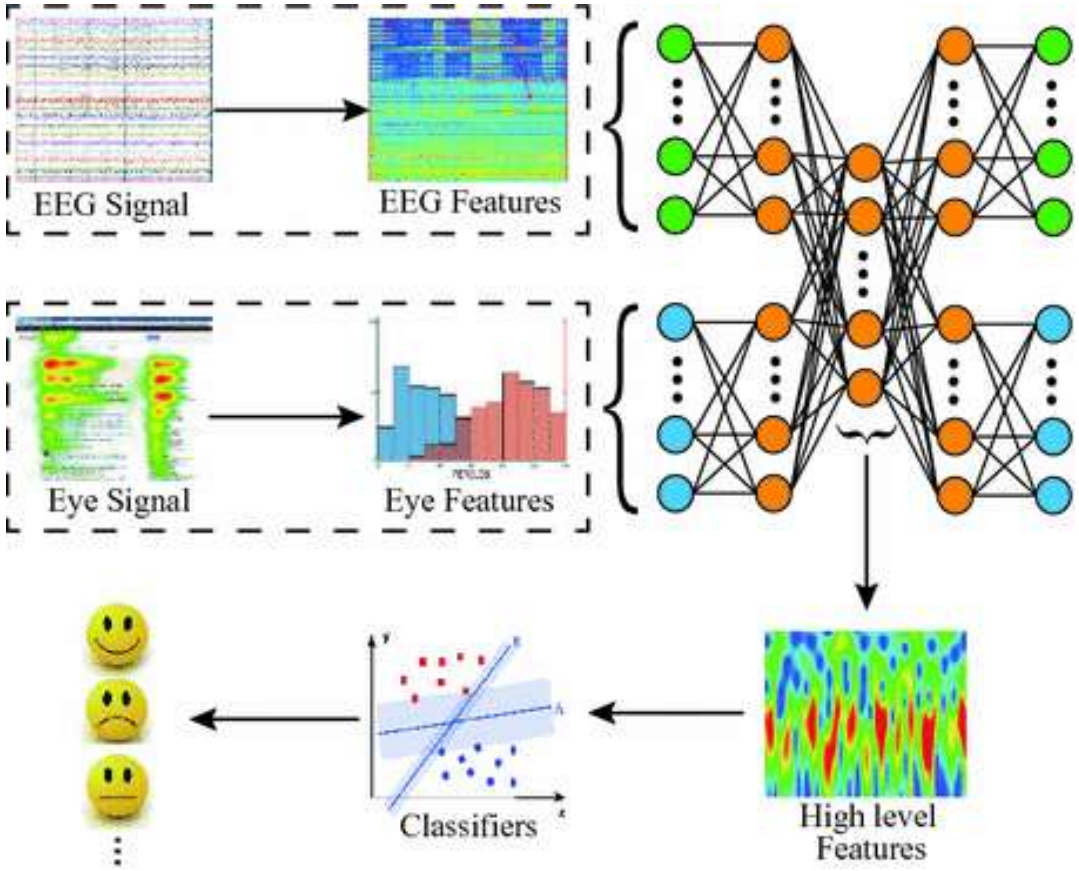
casa Paganini infomus

6. Multimodal machine learning

Multimodal Machine Learning

- A recent approach (Baltrusaitis et al., 2019) to the problem of putting together data from different modalities.
- Objective: building models to process information from multiple modalities.



Multimodal Machine Learning

- Five core technical open challenges (Baltrusaitis et al., 2019):
  - Representation: learning how to represent and summarize multimodal data in a way that exploits the complementarity and redundancy of multiple modalities.
  - Translation: how to translate data from one modality to another.
  - Alignment: how to identify the direct relations between elements from two or more modalities.
  - Fusion: how to join information from two or more modalities to perform a prediction.
  - Co-learning: how to transfer knowledge between modalities, their representations and predictive models.

Multimodal fusion

casa Paganini infomus

- The process of integrating information from various input modalities and combining them into a complete command [for a multimodal system] (D'Ulizia, 2009).
- Other terms are used in the literature:
  - Combination (Neal et al., 1989)
  - Cooperation of modalities (Martin et al., 1998)
  - Integration / multimodal integration (Pfleger, 2004; Shikler et al., 2004; Johnston and Bangalore, 2005).

Multimodal fusion: categories

casa Paganini infomus

- Two major categories:
  - Model-agnostic approaches: they do not depend on specific machine learning methods. They use techniques not designed to cope with multimodal data.
  - Model-based approaches: they address fusion in their construction. They include kernel-based methods, graphical models, and neural networks (Baltrusaitis et al., 2019).

A Summary of Our Taxonomy of Multimodal Fusion Approaches

Fusion type	Out	Temp	Task	Reference
Model-agnostic				
Early	class	no	Emotion rec.	[35]
Late	reg	yes	Emotion rec.	[175]
Hybrid	class	no	Multimedia event detection	[122]
Model-based				
Kernel-based	class	no	Object class.	[32], [69]
	class	no	Emotion rec.	[94], [189]
Graphical models	class	yes	AVSR	[78]
	reg	yes	Emotion rec.	[14]
	class	no	Media class.	[97]
Neural networks	class	yes	Emotion rec.	[100], [232]
	class	no	AVSR	[157]
	reg	yes	Emotion rec.	[39]

Examples of Multimodal Fusion Approaches.
Source: Baltrusaitis et al., 2019.

Model-agnostic approaches

- Feature level or early fusion: the features extracted from input data are first combined by a Feature Fusion unit (FF) and then sent to a single Analysis Unit (AU) that performs the analysis task (e.g., it makes a decision).

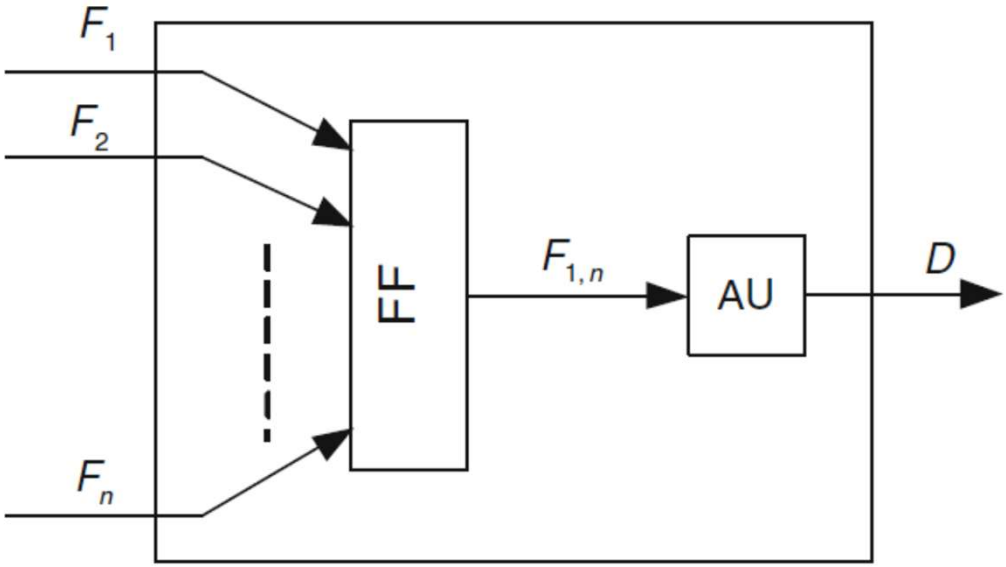


Figure from (Atrey et al., 2009)

Model-agnostic approaches

- Decision level or late fusion:
  - The Analysis Units (AUs) provide local decisions $D_1, \dots, D_N$ based on individual feature vectors $\boldsymbol{F}_1, \dots, \boldsymbol{F}_N$.
  - The local decisions are combined using a Decision Fusion (DF) unit to make a fused decision vector.
  - Such a decision vector is further analyzed by another Analysis Unit to get a final decision D.

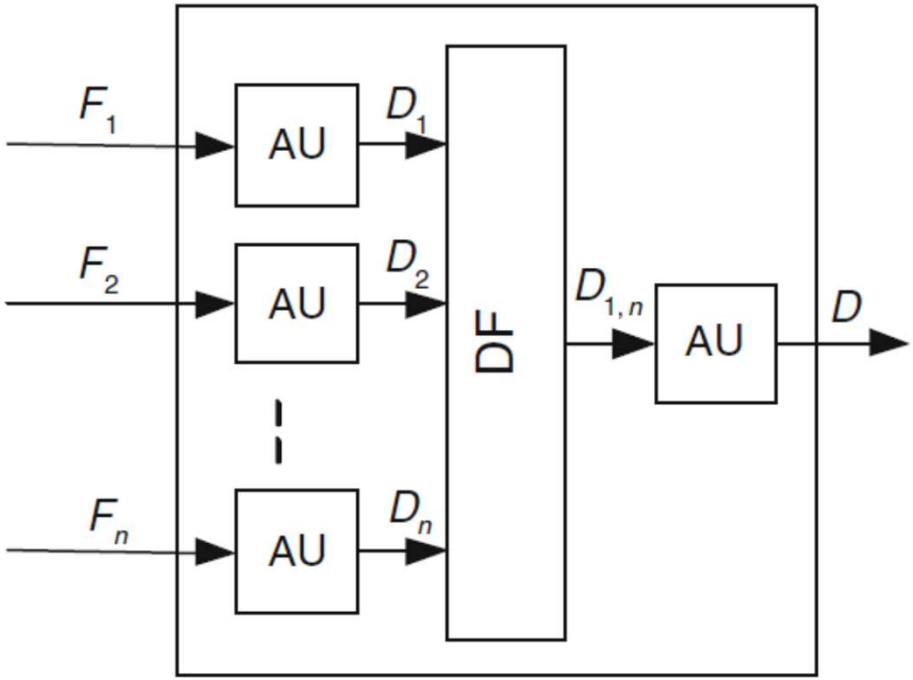


Figure from (Atrey et al., 2009)

Model-agnostic approaches

- Hybrid fusion is a combination of early and late fusion:
  - The features are first fused by an FF unit and then the feature vector is analyzed by an AU.
  - Other individual features are analyzed by different AUs and decisions are fused by a DF unit.
  - Finally, all the decisions obtained from the previous stages are further fused by a DF to obtain the final decision.

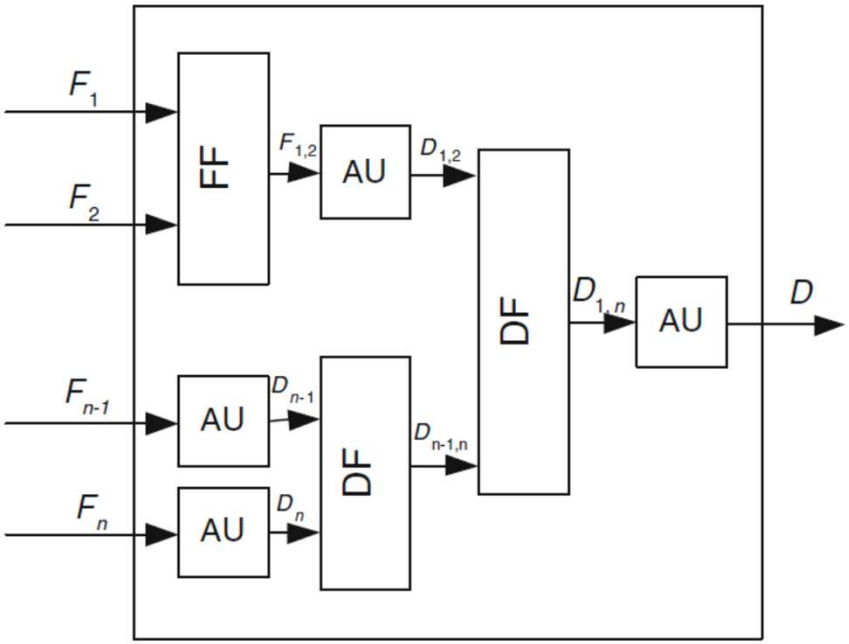


Figure from (Atrey et al., 2009)

Fusion methods

- These include statistical rule-based methods such as max, min, and, or, majority voting, and linear weighted fusion.
- Max, min, and, or apply these simple operators mainly for feature level fusion (e.g., select the feature having the maximum value at each given time).
- Majority voting is mainly used in decision level fusion. That is, the final decision is the one made by the majority of the classifiers.

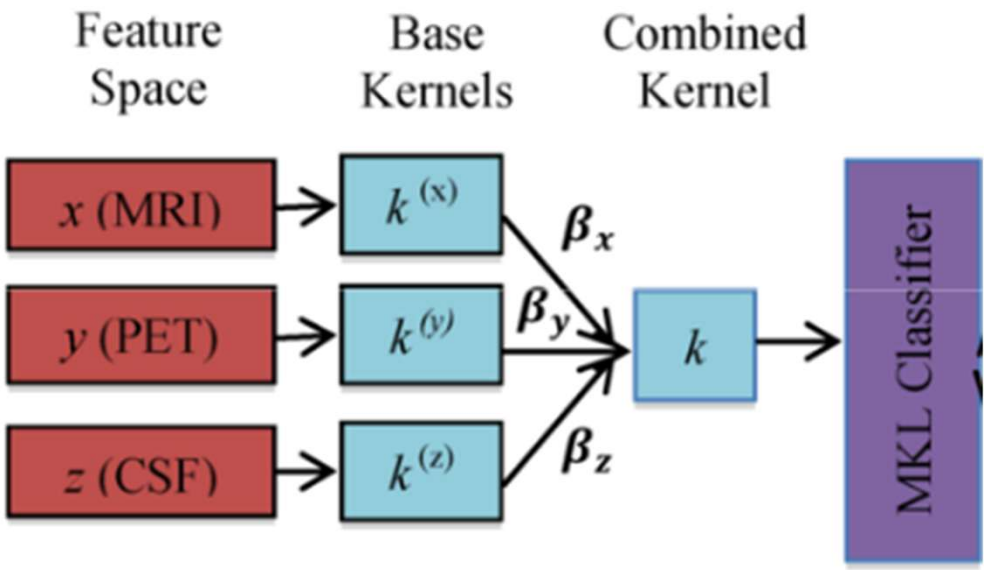
Early / late fusion: pros and cons

- Early fusion:
  - Pros: easy to implement (concatenation of features); it requires the training of a single model if used with machine learning.
  - Cons: it can end up very high dimensional; more difficult to use if features have different framerates.
- Late fusion:
  - Pros: different models can be used for each modality; it is easy to make predictions when one or more modalities are missing.
  - Cons: it does not model low level interactions between modalities.
- Hybrid fusion:
  - It combines benefits of both early and late fusion approaches.

Model-based approaches

casaPaganini infomus

- An example: Multiple Kernel Learning (MLK):
  - Extensions of SVM.
  - Multiple kernels are used instead of selecting one specific kernel function and its corresponding parameters. Then, a combination of these kernel is used.



Model-based approaches

- Multiple kernel learning (MLK):
  - Pros: broad applicability in various domains and across different modalities, loss function is convex, the approach can be applied to both classification and regression.
  - Cons: reliance on training data (support vectors) during test.
- Another common choice nowadays is Neural Networks: they enable to learn from large amount of data and end-to-end training of both multimodal representation and fusion. They can also learn complex decision boundaries. But they suffer of lack of interpretability and need large training datasets.

Model-based approaches

casa Paganini infomus

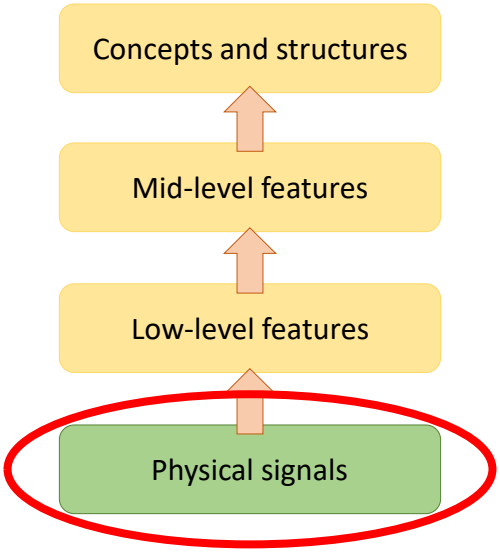
- Other examples (Atrey et al., 2009):

Fusion method	Level of fusion	The work	Modalities	Multimedia analysis task
Support vector machine	Decision	Adams et al. [3]	Video (color, structure, and shape), audio (MFCC) and textual cues	Semantic concept detection
		Aguilar et al. [4]	Fingerprint, signature, face (MCYT Multimodal Database, XM2VTS face database)	Biometric verification
		Iyenger et al. [58]	Audio, video	Semantic concept detection
		Wu et al. [141]	Color histogram, edge orientation histogram, color correlogram, co-occurrence texture, motion vector histogram, visual perception texture, and speech	Semantic concept detection
		Bredin and Chollet [19]	Audio (MFCC), video (DCT of lip area), audio-visual speech synchrony	Biometric identification of talking face
	Hybrid	Wu et al. [143]	Video, audio	Multimedia data analysis
		Zhu et al. [156]	Image (low-level visual features, text color, size, location, edge density, brightness, contrast)	Image classification
		Ayache et al. [11]	Visual, text cue	Semantic indexing
Bayesian inference	Feature	Pitsikalis et al. [102]	Audio (MFCC), video (Shape and texture)	Speech recognition
	Decision	Meyer et al. [85]	Audio (MFCC) and video (lips contour)	Spoken digit recognition
	Hybrid	Xu and Chua [149]	Audio, video, text, web log	Sports video analysis
		Atrey et al. [8]	Audio (ZCR, LPC, LFCC) and video (blob location and area)	Event detection for surveillance
Dempster–Shafer theory	Feature	Mena and Malpica [84]	Video (trajectory coordinates)	Segmentation of satellite images
	Decision	Guironnet et al. [44]	Audio (phonemes) and visual (visemes)	Video classification
		Singh et al. [116]	Audio (MFCC), video (DCT of the face region) and the synchrony score	Finger print classification
		Reddy [110]	Audio (MFCC), video (Eigenlip)	Human computer interaction
	Hybrid	Bendjebbour et al. [16]	Video (trajectory coordinates)	Segmentation of satellite images
Dynamic Bayesian networks	Feature	Wang et al. [138]	Audio (cepstral vector), visual (gray-level histogram difference and motion features)	Video shot classification
		Nefian et al. [86]	Audio (MFCC) and visual (2D-DCT coefficients of the lips region)	Speech recognition
		Nock et al. [90, 91]	Audio (MFCC) and video (DCT coefficients of the lips region	Speaker localization
		Chaisorn et al. [25]	Audio (MFCCs and perceptual features), video (color, face, video-text, motion)	Story segmentation in news video
		Adams et al. [3]	Video (color, structure, and shape),audio (MFCC) and textual cues	Video shot classification
		Beal et al. [15]	Audio and video—the details of features not available	Object tracking
		Bengio et al. [17]	Speech (MFCC) and video (shape and intensity features)	Biometric identity verification
		Hershey et al. [46]	Audio (Spectral components), video (fine-scale appearance and location of the lips)	Speaker localization
		Zou and Bhanu [158], Noulas and Krose [92]	Audio (MFCC) and video (pixel value variation)	Human tracking
		Ding and Fan [38]	Video (spatial color distribution and the angle of yard lines)	Shot classification in a sports video

5. Speech

1

A conceptual framework

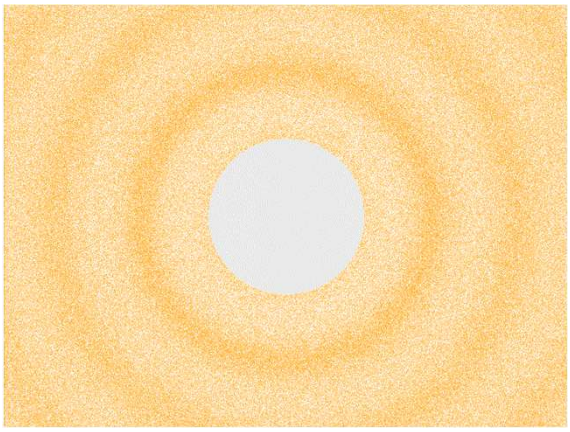


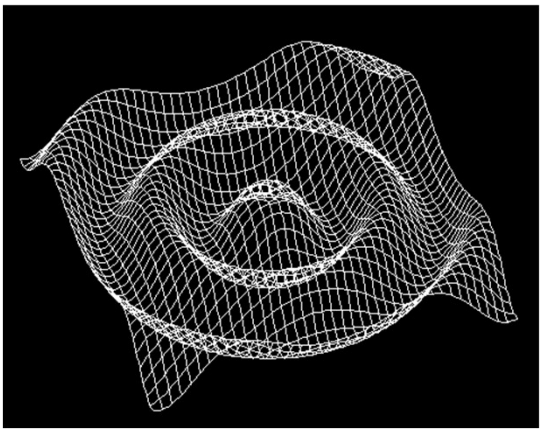
2

Sound

casa Paganini infomus

- Sound is a vibration that typically propagates as an audible wave of pressure.



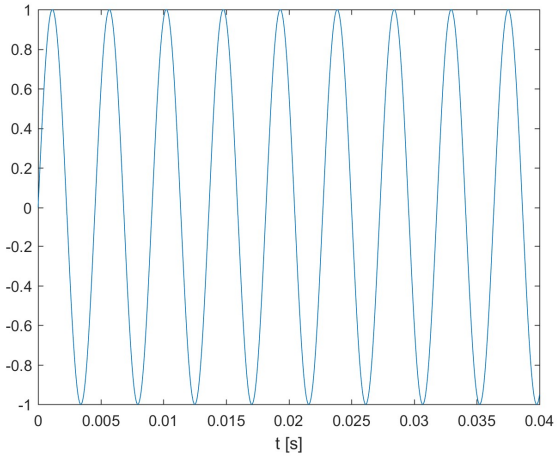


3

Pure and complex sounds

casa Paganini infomus

- Pure sound: $s_P(t) = A \cdot \sin(2\pi f t + \varphi)$



A = 1

f = 220 Hz

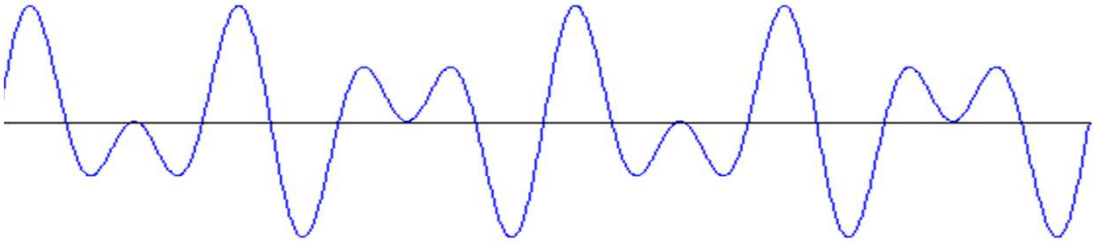
φ = 0

4

Pure and complex sounds

casa Paganini infomus

- Most sounds are instead complex sounds, which are a combination of many sound waves having different frequencies and summed all together.
- Almost all natural sounds are indeed complex sounds.



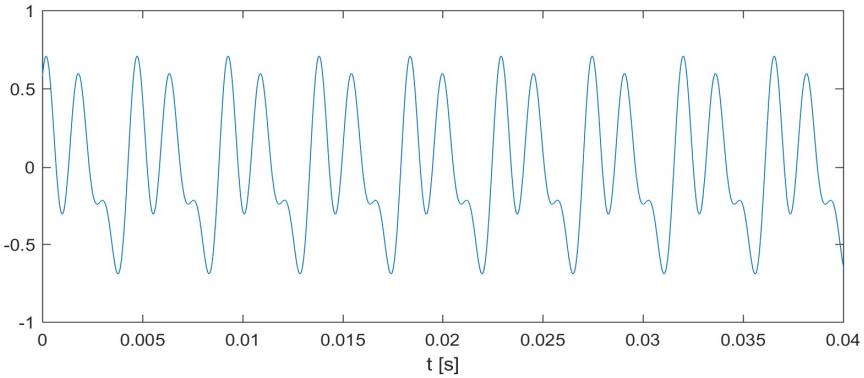
5

Pure and complex sounds

casa Paganini infomus

- Complex sound:

$$s_C(t) = A_1 \cdot \sin(2\pi f_1 t + \varphi_1) + A_2 \cdot \sin(2\pi f_2 t + \varphi_2) + A_3 \cdot \sin(2\pi f_3 t + \varphi_3) + \ldots + A_N \cdot \sin(2\pi f_N t + \varphi_N)$$



N = 3,
$A_1 = A_2 =$
$= A_3 = 1/3$
$f_1 = 220$ Hz
$f_2 = 440$ Hz
$f_3 = 880$ Hz
$\varphi_1 = 0$,
$\varphi_2 = \pi/2$,
$\varphi_3 = \pi/4$

6

Complex sounds

casa Paganini infomus

- Complex sounds can be distinguished in:
  - Periodic sound waves
  - Non periodic sound waves
- According to French mathematician and physicist Jean Baptiste Joseph Fourier, any periodic sound wave can be decomposed in the sum of (possibly infinite) pure sounds whose frequencies are multiples of the frequency corresponding to the period of the sound wave (harmonic series).

Jean-Baptiste Joseph Fourier (1768 – 1830)

7

Complex sounds

casa Paganini infomus

- This means that a periodic complex sound can be written as:

$$s_H(t) = A_0 + A_1 \cdot \sin(2\pi f_0 t + \varphi_1) + A_2 \cdot \sin(2\pi(2f_0)t + \varphi_2) + A_3 \cdot \sin(2\pi(3f_0)t + \varphi_3) + A_4 \cdot \sin(2\pi(4f_0)t + \varphi_4) + \ldots$$

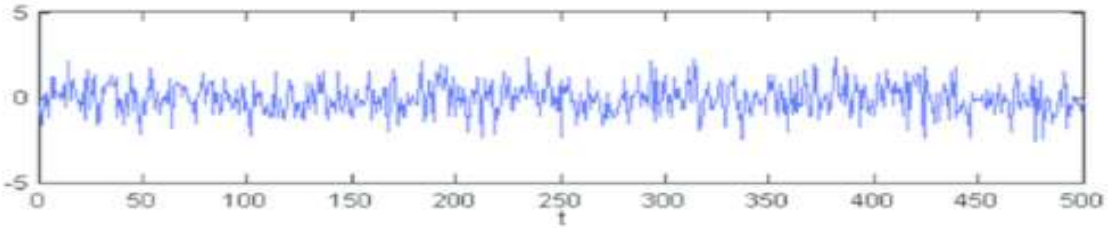
- Frequency f_0, corresponding to the period (i.e., $\mathrm{T} = 1/f_0$), is called fundamental frequency of the sound.
- The other frequency components are called harmonics.

8

Complex sounds

casa Paganini infomus

- Non periodic sounds also consists of a sum of pure sounds.
- But the frequencies of the composing pure sounds are not bound to comply with any particular relationship between them, i.e., they do not have to be multiple of any frequency.
- Such frequency components are generally called partials and are usually infinite.

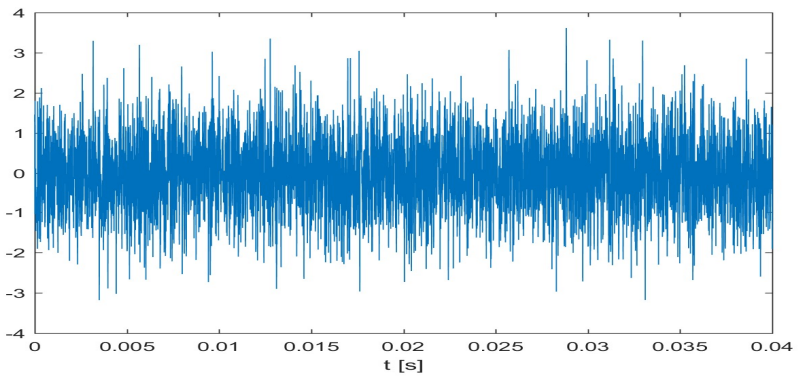


9

Complex sounds

casa Paganini infomus

- For example, white noise is a non-periodic sound having equal intensity at different frequencies, and ideally containing all possible frequencies.

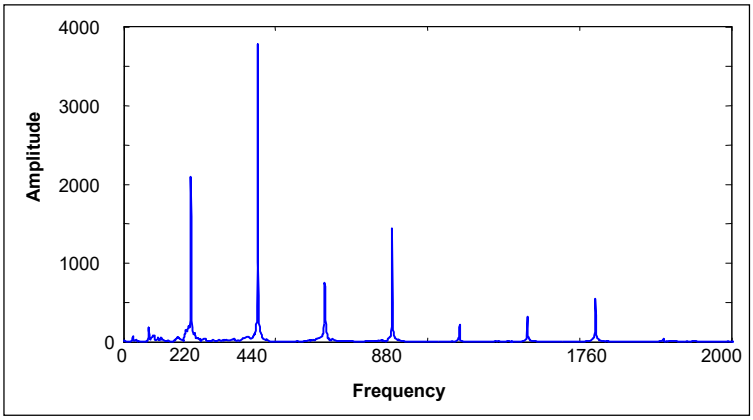


10

Frequency spectrum

casa Paganini infomus

- The frequency spectrum of a periodic sound wave, i.e., of a harmonic sound, displays vertical lines in correspondence of the frequencies of its harmonics.

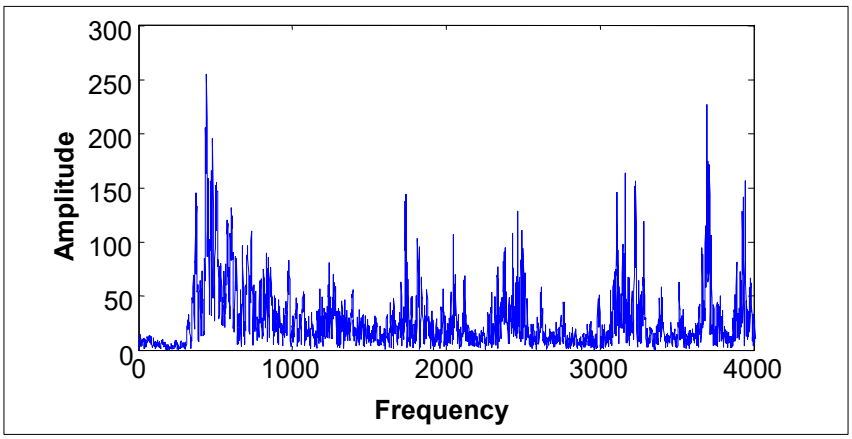


11

Frequency spectrum

casa Paganini infomus

- The frequency spectrum of a non-periodic sound usually does not display any prominent partial.



12

Phones

- Speech sounds are called phones.
- Phones serves as basic units of phonetic speech analysis.
- The pronunciation of a word can thus be represented as a string of phones.
- The standard phonetic representation for transcribing languages is the International Phonetic Alphabet (IPA).
- A simpler option is ARPAbet that uses ASCII symbols to represent an American-English subset of the IPA.
- A phonetic transcription (i.e., based on phones) is enclosed within square brackets ([ ]).

13

IPA and ARPAbet

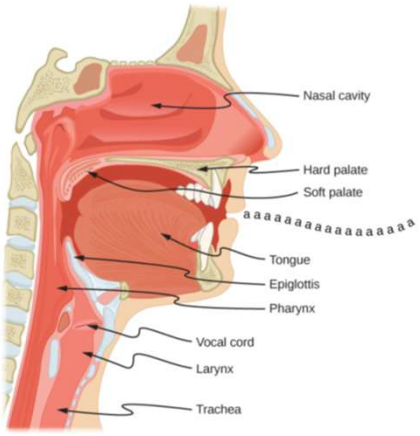
ARPAbet Symbol	IPA Symbol	Word	ARPAbet Transcription	ARPAbet Symbol	IPA Symbol	Word	ARPAbet Transcription
[p]	[p]	parsley	[p aa r s l iy]	[iy]	[i]	lily	[l ih l iy]
[t]	[t]	tea	[t iy]	[ih]	[ɪ]	lily	[l ih l iy]
[k]	[k]	cook	[k uh k]	[ey]	[eɪ]	daisy	[d ey z iy]
[b]	[b]	bay	[b ey]	[eh]	[ɛ]	pen	[p eh n]
[d]	[d]	dill	[d ih l]	[ae]	[æ]	aster	[ae s t axr]
[g]	[g]	garlic	[g aa r l ix k]	[aa]	[ɑ]	poppy	[p aa p iy]
[m]	[m]	mint	[m ih n t]	[ao]	[ɔ]	orchid	[ao r k ix d]
[n]	[n]	nutmeg	[n ah t m eh g]	[uh]	[ʊ]	wood	[w uh d]
[ng]	[ŋ]	baking	[b ey k ix ng]	[ow]	[oʊ]	lotus	[l ow dx ax s]
[f]	[f]	flour	[f l aw axr]	[uw]	[u]	tulip	[t uw l ix p]
[v]	[v]	clove	[k l ow v]	[ah]	[ʌ]	butter	[b ah dx axr]
[th]	[θ]	thick	[th ih k]	[er]	[ɝ]	bird	[b er d]
[dh]	[ð]	those	[dh ow z]	[ay]	[aɪ]	iris	[ay r ix s]
[s]	[s]	soup	[s uw p]	[aw]	[aʊ]	flower	[f l aw axr]
[z]	[z]	eggs	[eh g z]	[oy]	[ɔɪ]	soil	[s oy l]
[sh]	[ʃ]	squash	[s k w aa sh]				
[zh]	[ʒ]	ambrosia	[ae m b r ow zh ax]				
[ch]	[tʃ]	cherry	[ch eh r iy]				
[jh]	[dʒ]	jar	[jh aa r]				
[l]	[l]	licorice	[l ih k axr ix sh]				
[w]	[w]	kiwi	[k iy w iy]				
[r]	[r]	rice	[r ay s]				
[y]	[j]	yellow	[y eh l ow]				
[h]	[h]	honey	[h ah n iy]				

ARPAbet and IPA symbols for English consonants (left) and vowels (right). Source: Jurafsky, D., and Martin, J. H., Speech and Language Processing.

14

Phones production

- Humans produce phones by expelling air from the lungs through the trachea, the larynx, and then out the mouth or nose.
- The larynx contains two small folds of muscle, the vocal folds.
- If the folds are close together, they vibrate as air passes through them; if they are far apart, they do not vibrate.



The vocal organs. Source OpenStax University Physics (CC BY 4.0).

15

Voiced and unvoiced sounds

- Sounds made with the vocal folds together and vibrating are called voiced sounds. Examples: [b], [d], [g], [v], [z], and all the English vowels.
- Sounds made without this sound cord vibration are called unvoiced or voiceless. Examples: [p], [t], [k], [f], [s].

[b]

[p]

16

Consonants and vowels

casa Paganini infomus

- Consonants are made by restriction or blocking of the airflow and can be voiced or unvoiced.
- Vowels have less obstruction, are usually voiced, and are generally louder and longer-lasting than consonants.

13 Vowels	IPA	ARPAbet	*Sound Spelling*
hate	eɪ	EY	ā
Pete	i	IY	ē
site	aɪ	AY	ī
note	oʊ	OW	ō
cute	u	UW	o͞o
hat	æ	AE	ă
pet	ɛ|e	EH	ĕ
sit	ɪ	IH	ĭ
not	ɔ|ɑ	AO or AA	ŏ
cut	ʌ|ə	AH or AX	ŭ
coin	ɔɪ	OY	oy
loud	aʊ	AW	ow
book	ʊ	UH	o͝o

24 Consonants	IPA	ARPAbet	*Sound Spelling*
buy	b	B	b
pie	p	P	p
die	d	D	d
tie	t	T	t
vie	v	V	v
fight	f	F	f
guy	g	G	g
kite	k	K	k
high	h	HH or H	h
joy	dʒ	JH	j
China	tʃ	CH	ch

24 Consonants	IPA	ARPAbet	*Sound Spelling*
lie	l	L	l
my	m	M	m
nigh	n	N	n
rye/turn	ɹ/ɪ	R	r/ur
zoo	z	Z	z
sigh	s	S	s
wise	w	W	w
yacht	j	Y	y
pleasure	ʒ	ZH	zh
shy	ʃ	SH	sh
they	ð	DH	dh
thigh	θ	TH	th
sing	ŋ	NX or NG	ng

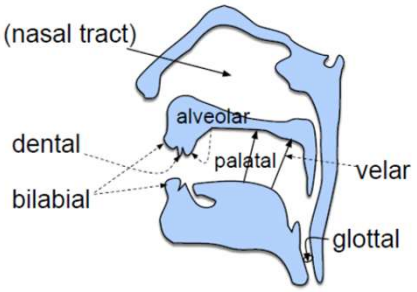
Consonants and vowels in English. Figure adapted from (Nakatsuka et al., 2020).

17

Place of articulation

casa Paganini infomus

- Consonants are distinguished by their point of maximum restriction of airflow (place of articulation):
  - Labial: lips coming together.
  - Dental: tongue against the teeth.
  - Alveolar: tongue against alveolar ridge.
  - Palatal: tongue against palate.
  - Velar: tongue against velum.
  - Glottal: vocal folds coming together.



Major English places of articulation. Source: Jurafsky, D., and Martin, J. H., Speech and Language Processing.

18

Manners of articulation

casa Paganini infomus

- Consonants are also distinguished by how the restriction in airflow is made (manner of articulation):
  - Stops (or plosives): airflow is completely blocked for a short time.
  - Nasals: air is allowed passing into the nasal cavity.
  - Fricatives: airflow is constricted but not cut off completely. Airflow is turbulent. Higher-pitched fricatives are called sibilants.
  - Approximants: articulators are close together but not close enough to cause turbulent airflow.
  - Taps or flaps: characterized by a quick motion of the tongue against the alveolar ridge.

19

Consonants

casa Paganini infomus

THE INTERNATIONAL PHONETIC ALPHABET (revised to 2020)

CONSONANTS (PULMONIC) ©①◎ 2020 IPA

	Bilabial	Labiodental	Dental	Alveolar	Postalveolar	Retroflex	Palatal	Velar	Uvular	Pharyngeal	Glottal
Plosive	p b			t d		ʈ ɖ	c ɟ	k g	q ɢ		ʔ
Nasal	m	ɱ		n		ɳ	ɲ	ŋ	ɴ		
Trill	ʙ			r					ʀ		
Tap or Flap		ⱱ		ɾ		ɽ					
Fricative	ɸ β	f v	θ ð	s z	ʃ ʒ	ʂ ʐ	ç ʝ	x ɣ	χ ʁ	ħ ʕ	h ɦ
Lateral fricative				ɬ ɮ							
Approximant		ʋ		ɹ		ɻ	j	ɰ			
Lateral approximant				l		ɭ	ʎ	ʟ			

Symbols to the right in a cell are voiced, to the left are voiceless. Shaded areas denote articulations judged impossible.

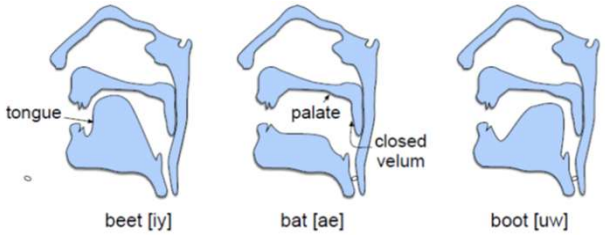
Chart of pulmonic consonants. Source: IPA.

20

Vowels

casa Paganini infomus

- Vowels are also characterized by position of the articulators:
  - Front vowels: tongue is raised toward the front.
  - Back vowels: tongue is raised toward the back.
  - High vowels: highest point of the tongue is comparatively high.
  - Mid vowels: highest point of the tongue is comparatively mid.
  - Low vowels: highest point of the tongue is comparatively low.

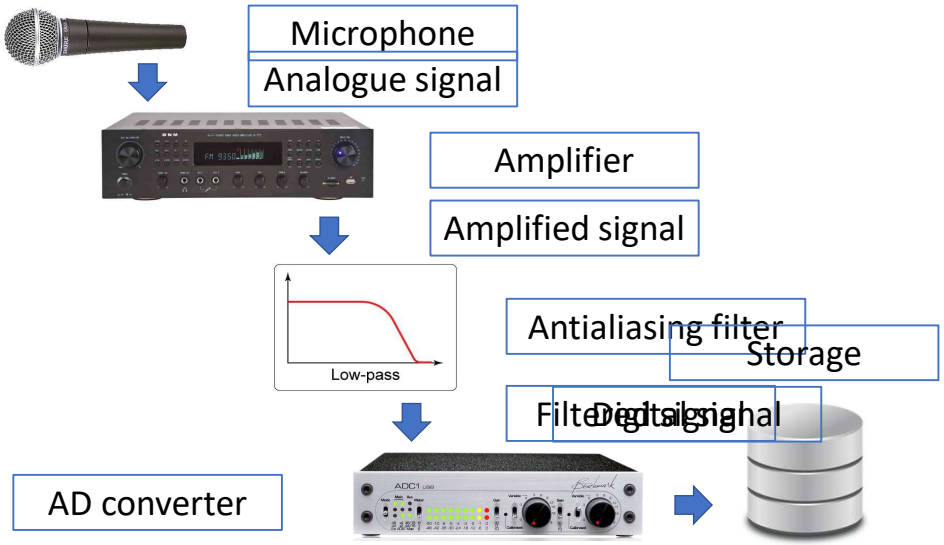


Tongue positions for English high front [iy], low front [ae] and high back [uw]. Source: Jurafsky, D., and Martin, J. H., Speech and Language Processing.

21

Digital recording of sound

casa Paganini infomus



22

Microphones

casa Paganini infomus

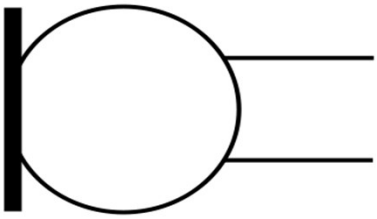
- Microphones capture variations in air pressure and transduce them into the variations of an electric signal (usually a voltage).
- Their sensitive transducer element is called element or capsule.
- Sound is converted to mechanical motion by means of a diaphragm, the motion of which is then converted to an electrical signal.

23

Microphones

casa Paganini infomus

- With respect to the transducer principle (i.e., how transduction is performed), common kinds of microphones are:
  - Carbon microphones.
  - Piezoelectric microphones.
  - Dynamic microphones
  - Ribbon microphones
  - Condenser and electret microphones.



24

Amplifiers

casa Paganini infomus

- The signal coming out from the microphone may need to be pre-amplified and amplified.
- An amplifier is an electronic device that can increase the amplitude of a signal.
- The amount of amplification provided by an amplifier is measured by its gain: the ratio of output to input.
- In an ideal amplifier, the output signal is a replica (of course with a higher amplitude) of the input signal.
- Real amplifiers are usually not able to exactly replicate the shape of the input signal, and so they introduce distortions.

25

Analog-to-Digital conversion

casa Paganini infomus

- Sound is transduced by a microphone to an analog signal as for analog recording, and possibly amplified.
- The analog signal is filtered and converted to a digital signal, through an Analog-to-Digital Converter (ADC). This can be either integrated into the recorder or separate.
- Conversion is operated by applying three basic operations: sampling, quantization, and coding.
- The resulting sound samples are stored in a digital storing device, a hard disk for example.

26

Exemplar parameters

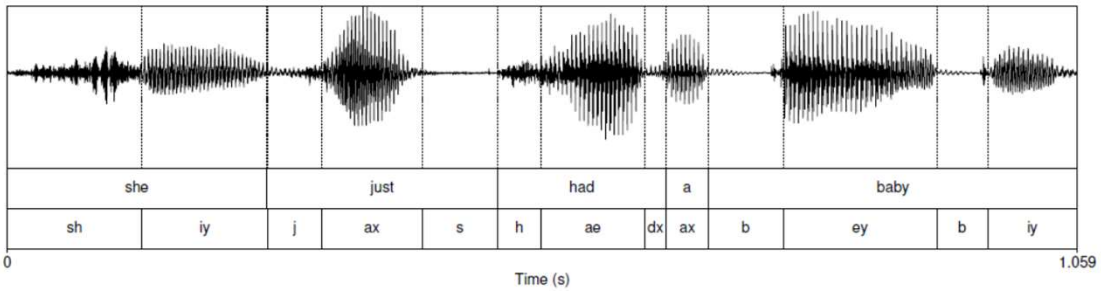
casa Paganini infomus

Quality	Sample Rate (KHz)	Bits per Sample	Mono/ Stereo	Data Rate (uncompressed) (kB/sec)	Frequency Band (KHz)
Telephone	8	8	Mono	8	0.200-3.4
AM Radio	11.025	8	Mono	11.0	0.1-5.5
FM Radio	22.05	16	Stereo	88.2	0.02-11
CD	44.1	16	Stereo	176.4	0.005-20
DAT	48	16	Stereo	192.0	0.005-20
DVD Audio	192 (max)	24 (max)	6 channels	1,200.0 (max)	0-96 (max)

27

An example

casa Paganini infomus

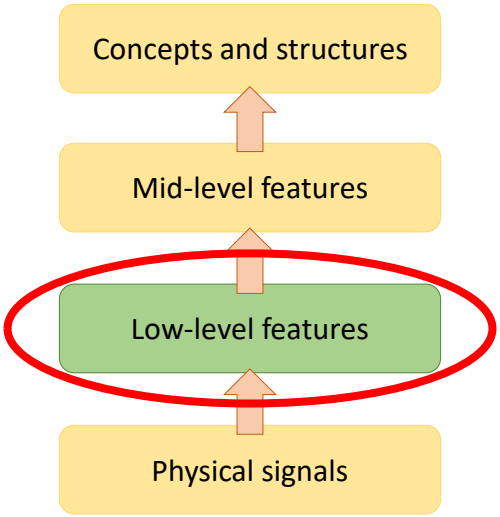


A waveform of the sentence “She just had a baby”.
Source: Jurafsky, D., and Martin, J. H., Speech and Language Processing.

28

A conceptual framework

casa Paganini infomus

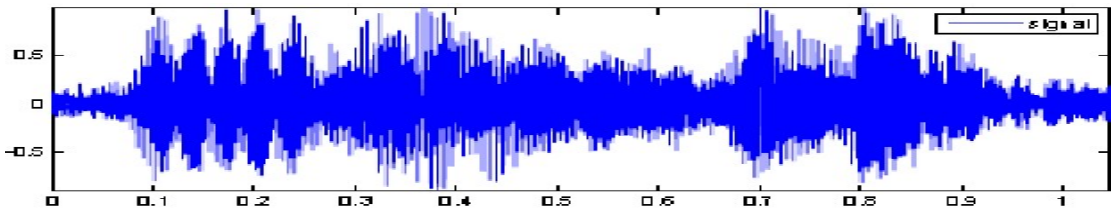


29

Audio frames

casa Paganini infomus

- The analysis of a whole temporal signal leads to features that represent the average content of the signal.
- To consider the dynamics of the features, analysis is carried out on a short-term window moving chronologically along the temporal signal.
- Each position of the window is called a frame.



30

Audio frames

casa Paganini infomus

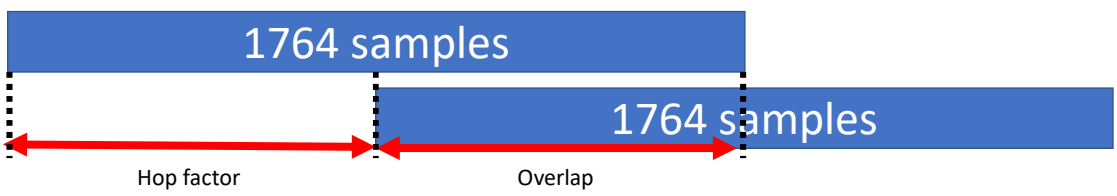
- Features of an audio frame:
  - Time duration of the frame (frame length), usually expressed in seconds or in milliseconds or as the number of samples that belong to the frame.
  - Hop factor, i.e., the distance, in seconds or milliseconds or as the number of frames, between consecutive frames.
  - If the hop factor is shorter than the time duration, overlapping occurs, that is frames are overlapped.
  - Hop factor and overlapping determine the sample rate of the computed sound features. Half-frame overlapping is often used.
  - Frame rate: number of frames per second (fps).

31

Audio frames

casa Paganini infomus

- An example:
  - Audio sampling rate: 44100 Hz.
  - Frame time duration: 40 ms, i.e., 44100 * 0.04 = 1764 samples.
  - Hop factor: 20 ms, i.e., 882 samples (half overlapping).
  - Sample rate of the computed features (one feature sample is computed for each sound frame): 1 / 20 ms = 50 Hz.
  - Number of frames per seconds = frame rate = 50 Hz.



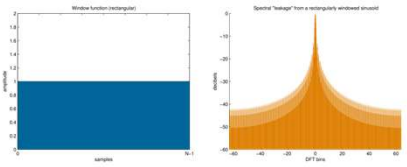
32

Windows

casa Paganini infomus

- Segmenting an audio signal in frames is equivalent to applying a rectangular window:

$$w_R(n) = \begin{cases} 1 & n = 0, \ldots, R-1 \\ 0 & elsewhere \end{cases}$$



- The rectangular window gives poor results, especially when analysis in the frequency domain is performed (i.e., the Fourier transform is computed).
- More sophisticated windowing function are thus used, especially when FFT is involved.

33

Windows

casa Paganini infomus

- Frequently used functions:
  - Hamming window

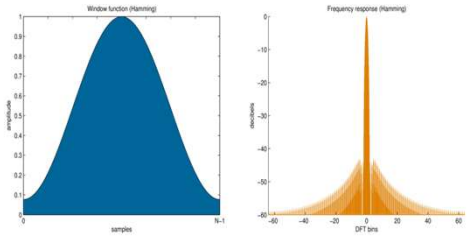
$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right)$$

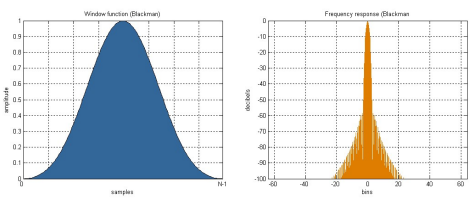
  - Blackman windows

$$w(n) = a_0 - a_1 \cos\left(\frac{2\pi n}{N-1}\right) + a_2 \cos\left(\frac{4\pi n}{N-1}\right)$$

$$a_0 = \frac{1-\alpha}{2}; \quad a_1 = \frac{1}{2}; \quad a_2 = \frac{\alpha}{2}$$

By common convention, the unqualified term Blackman window refers to α = 0.16





34

Low-level features

- The simplest descriptors of a sound wave are those related to its amplitude and (fundamental) frequency.
- The most commonly used features for automatic speech recognition are, however, the log mel spectral coefficients and the Mel-Frequency Cepstral Coefficients (MFCCs).
- Other sets of features were developed for specific tasks: for example, the Geneva Minimalistic Acoustic Parameter Set (GeMAPS) is used for applications in affective computing.

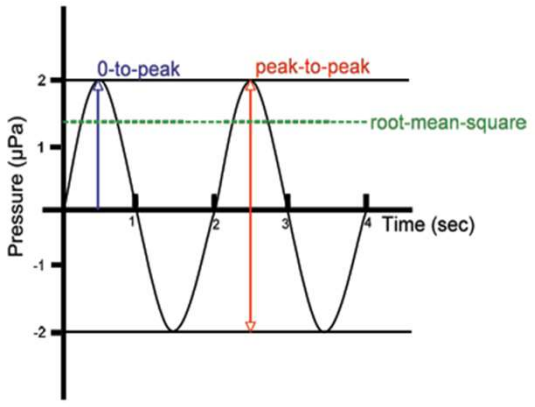
35

Amplitude

- The most commonly used measure for amplitude is Root Mean Square (RMS). This is defined as:

$$RMS_n = \sqrt{\frac{1}{N}\sum_{i=n-N+1}^{n} s_i^2}$$

where N is the selected size of the audio frame.



Commonly used measures for amplitude.
Source: https://dosits.org/science/advanced-topics/introduction-to-signal-levels/

36

Frequency

casa Paganini infomus

- The autocorrelation function (AR) for signal s_i is defined as:

$$R(k) = \sum_{i=-\infty}^{+\infty} s_i \cdot s_{i+k}$$

- The short-time autocorrelation function for an audio frame of N samples, is computed as:

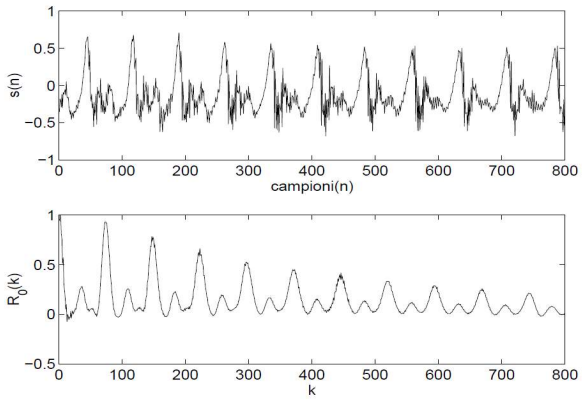
$$R_n(k) = \sum_{i=n-N+1}^{n-k} s_i \cdot s_{i+k}$$

37

Short-Time Autocorrelation

casa Paganini infomus

- If s_i is periodic with period T, its autocorrelation function is periodic with the same period, i.e., $R(k) = R(k+T)$, and has maxima in correspondence of T, $2T$, …



Example: Autocorrelation of a voiced sound (Figure by Mion, De Poli)

38

F0 estimation

casa Paganini infomus

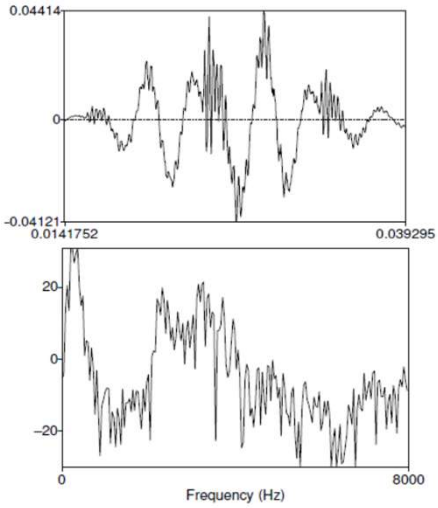
- Fundamental frequency can be estimated from the AR function as follows:
  - Pre-processing: filtering and denoising.
  - Extraction of the candidate period: k^* corresponding to the first maximum of the autocorrelation function, after the maximum at $k = 0$.
  - Computation of the fundamental frequency: $\mathrm{F0} = f_s/k^*$.
  - Post-processing: fine-tuning and correction of possible errors.

39

Log mel spectral features

casa Paganini infomus

- Log mel spectral features represent the amount of energy at different frequency bands.
- The first step to compute them consists of extracting spectral information: the discrete Fourier transform (DFT) is thus applied to the windowed signal.
- A commonly used algorithm for computing the DFT of a signal is the fast Fourier transform (FFT).

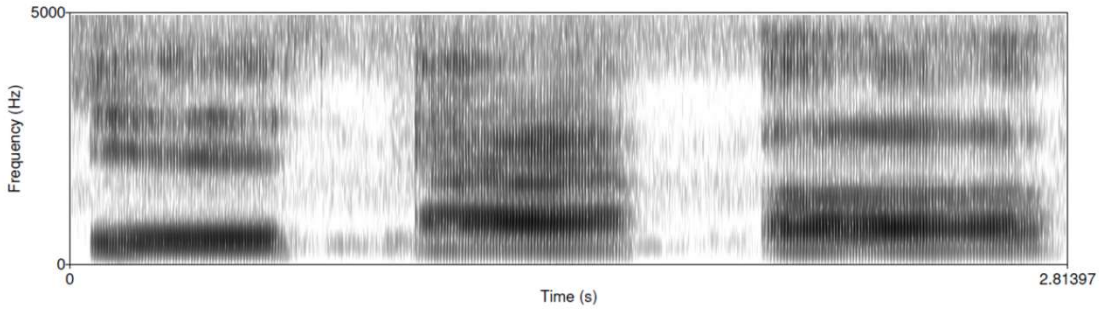


A 25 ms Hamming-windowed portion of a signal (vowel [iy]) and its magnitude spectrum. Source: Jurafsky, D., and Martin, J. H., Speech and Language Processing.

40

Spectrogram

- While a spectrum shows the frequency components of a wave at one point in time, a spectrogram is a way of envisioning how the different frequencies that make up a waveform change over time.

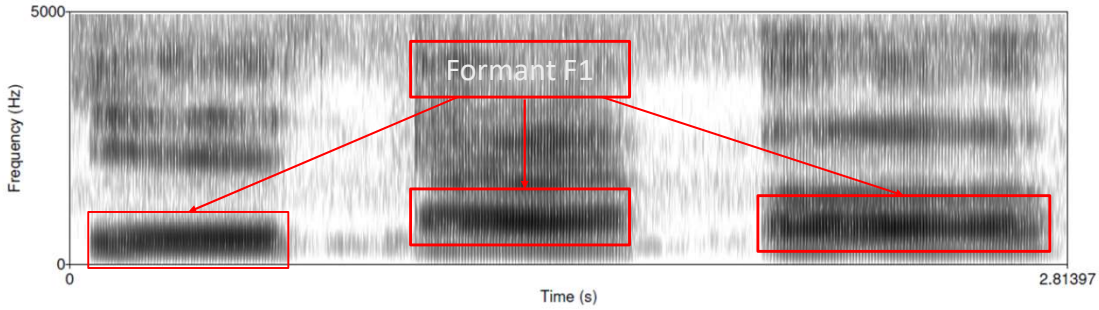


Spectrograms for three American English vowels, [ih], [ae], and [uh].
Source: Jurafsky, D., and Martin, J. H., Speech and Language Processing.

41

Formants

- Each dark bar (or spectral peak) is called a formant: this is a frequency band particularly amplified by the vocal tract.
- Since different vowels are produced with the vocal tract in different positions, they display different resonances.



Spectrograms for three American English vowels, [ih], [ae], and [uh].
Source: Jurafsky, D., and Martin, J. H., Speech and Language Processing.

42

Mel scale

casa Paganini infomus

- Human hearing, is not equally sensitive at all frequency bands; it is less sensitive at higher frequencies.
- This bias helps human recognition, since information in low frequencies is crucial for distinguishing phones, while information in high frequencies is less crucial.
- To model this human property, the mel scale (after the word *melody*) is applied.
- Pairs of sounds that are perceptually equidistant are separated by an equal number of mels.

43

Mel scale

casa Paganini infomus

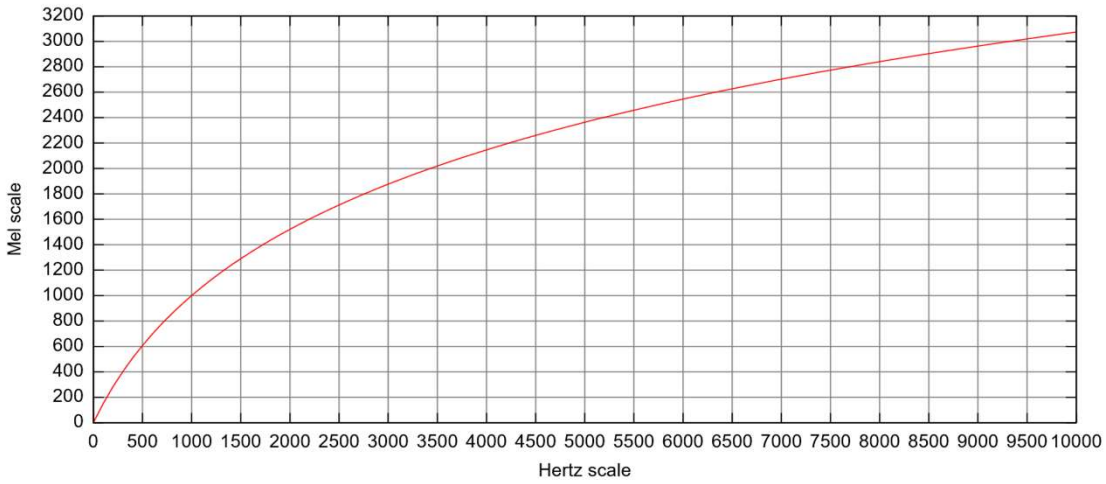
- The reference point between the mel scale and normal frequency measurement is defined by assigning 1000 mels to a 1000 Hz tone (40 dB above the listener's threshold).
- That is, mels are obtained from Hertz as follows:

$$mel(f) = 2595 \log_{10}\left(1 + \frac{f}{700}\right) = 1127 \ln\left(1 + \frac{f}{700}\right)$$

- Thus, for example, 440Hz corresponds to 549.64 mels.

44

Mel scale

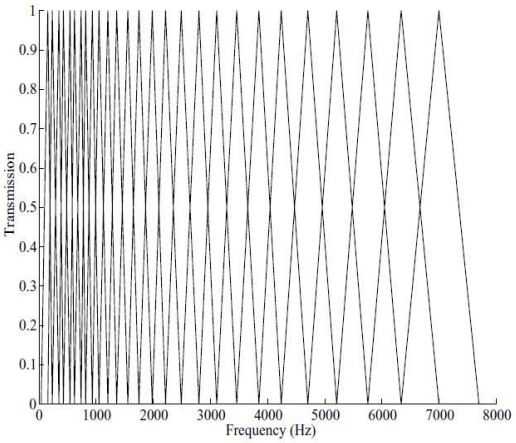


Transformation from Hz to mel: on the mel scale, 1000 Hz correspond to 1000 mel
(source: Wikipedia, author: Krishna Vedala)

45

Mel filter bank

- This intuition is implemented by passing the signal through a bank of filters that collect energy from each frequency band, spread logarithmically so that we have very fine resolution at low frequencies, and less resolution at high frequencies.



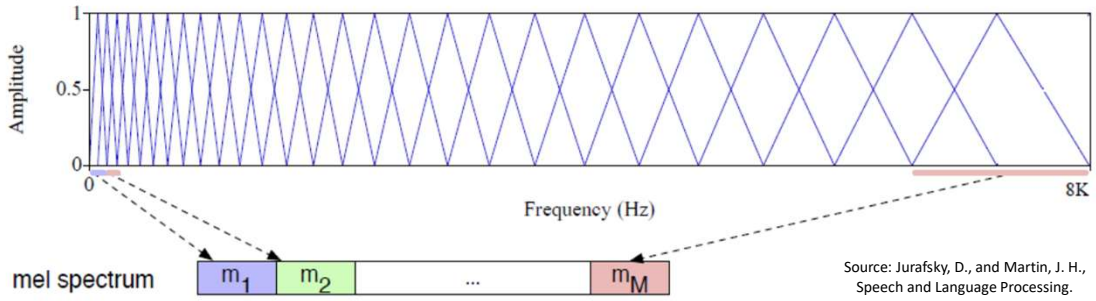
Mel-scale filterbank. Figure by Mion, De Poli

46

Mel spectrum

casa Paganini infomus

- A mel spectrum is obtained by multiplying the spectrum by the transfer function of each filter in the mel filter bank.
- The spectral values output from the mel filter bank are then summed up, and finally the logarithm of each value is taken.



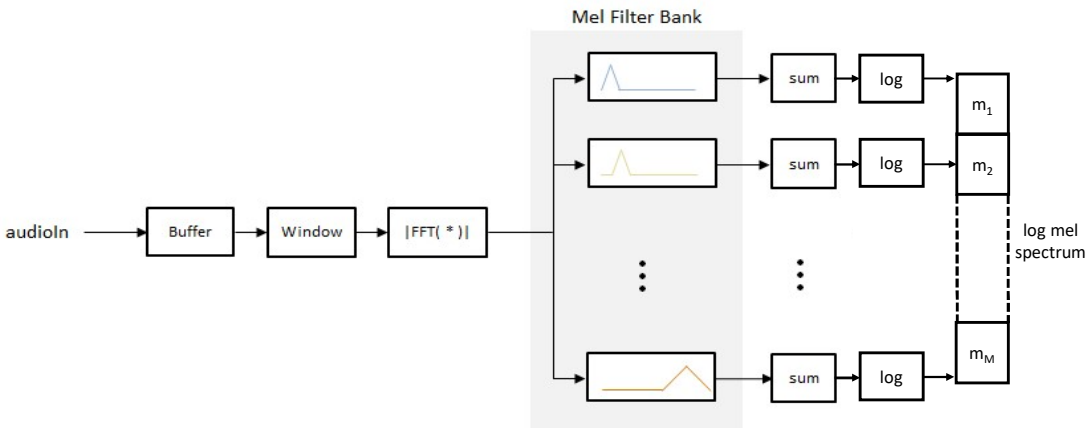
Source: Jurafsky, D., and Martin, J. H., Speech and Language Processing.

47

Log mel spectral features

casa Paganini infomus

- The whole process can be summarized as follows:



Adapted from: MathWorks, https://it.mathworks.com/help/audio/ref/melspectrogram.html

48

Cepstral analysis

casa Paganini infomus

- Peaks in the spectrum correspond to dominant frequency components and may denote formants in a speech signal.
- A representation of the spectral envelop can thus represent an alternative to the log mel spectrum, to use as features.

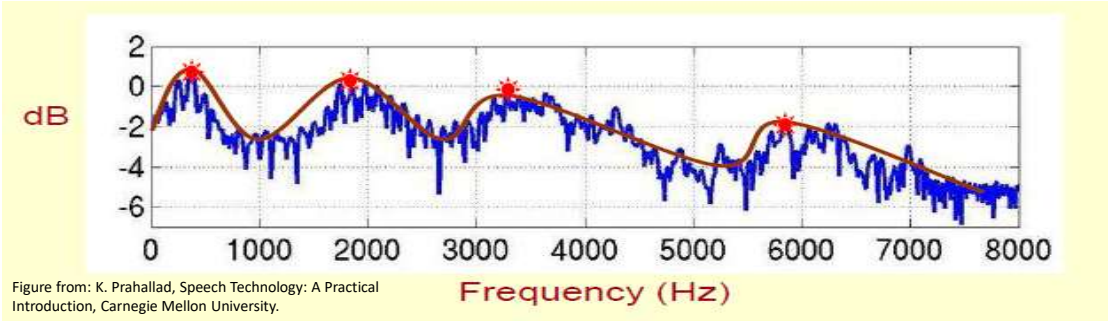


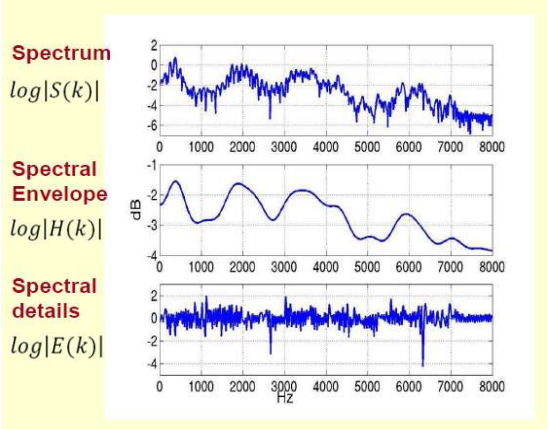
Figure from: K. Prahallad, Speech Technology: A Practical Introduction, Carnegie Mellon University.

49

Cepstral analysis

casa Paganini infomus

- The goal is to separate spectral envelope from spectral details.



$$\log|S_k| = \log|H_k| + \log|E_k|$$

Figure from: K. Prahallad, Speech Technology: A Practical Introduction, Carnegie Mellon University.

50

Cepstral analysis

casa Paganini infomus

- Cepstral analysis is a mathematical trick consisting of:
  - Considering the spectrum as it were a signal in the time domain;
  - Computing the magnitude of the DFT of the logarithm of the spectrum magnitude, i.e., the signal cepstrum. Note that since cepstrum is computed from a real only function, both DFT and inverse DFT produce the same shape result:

$$c_n = DFT(\log|S_k|) = DFT^{-1}(\log|S_k|)$$

  - Applying a low-pass filter to the separate spectral envelope from the spectral details.

51

Cepstral analysis

casa Paganini infomus

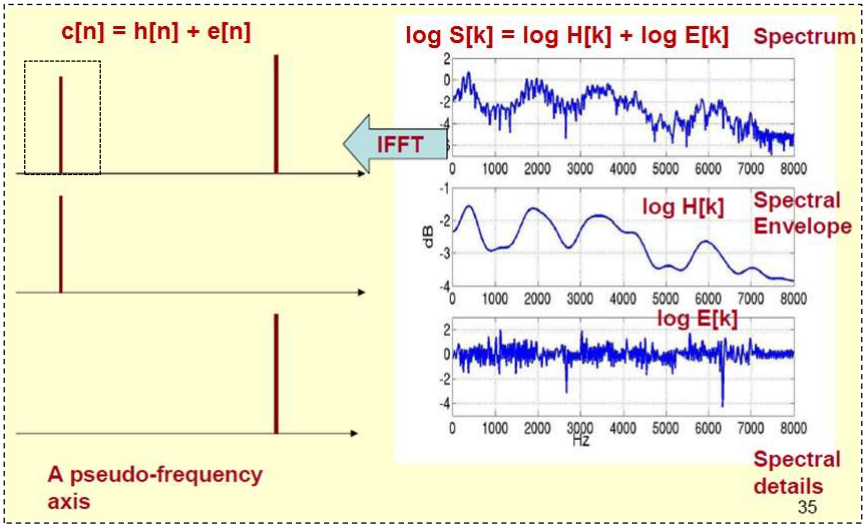


Figure from: K. Prahallad, Speech Technology: A Practical Introduction, Carnegie Mellon University.

52

Cepstral analysis

casa Paganini infomus

- Given a low-pass window in the cepstral domain, defined for example as:

$$w_n^{LP} = \begin{cases} 1 & |n| < n_c \\ 0.5 & |n| = n_c \\ 0 & |n| > n_c \end{cases}$$

the spectral envelope can be obtained as:

$$\log|H_k| = DFT\left(w_n^{LP} \cdot c_n\right) = DFT\left(w_n^{LP} \cdot DFT^{-1}(\log|S_k|)\right)$$

- Note that if DFT^{-1} is used to compute cepstrum, then DFT is used to go back to the spectrum.

53

Cepstral analysis

casa Paganini infomus

- Cepstrum analysis has developed its own terminology for its most important aspects.
- Cepstrum is an anagram of spectrum, derived by reversing the first four letters of “spectrum”.
- The independent variable of a cepstral graph is called the quefrency (an anagram of frequency). This a measure of time, though not in the sense of a signal in the time domain.
- Filtering in the cepstral domain is sometimes called liftering.

54

MFCCs

casa Paganini infomus

- The results of Mel-Cepstral analysis is a collection of coefficients, which are called are called Mel-Frequency Cepstral Coefficients (MFCCs), that describe the envelope of the spectrum.
- MFCCs are obtained from the log mel spectrum by:
  - Taking the discrete cosine transform (DCT) of the list of the mel log spectrum, as if it were a signal over time. DCT is used instead of DFT for operating a compression of the size of data.
  - Applying liftering to the result, that is taking n_C values from the output of the DCT (from the first value up to the n_C-th one).

55

MFCCs

casa Paganini infomus

- The whole process is summarized as follows:

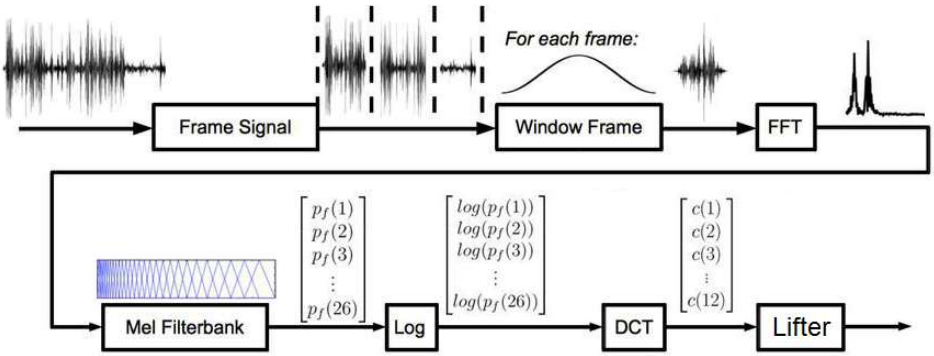


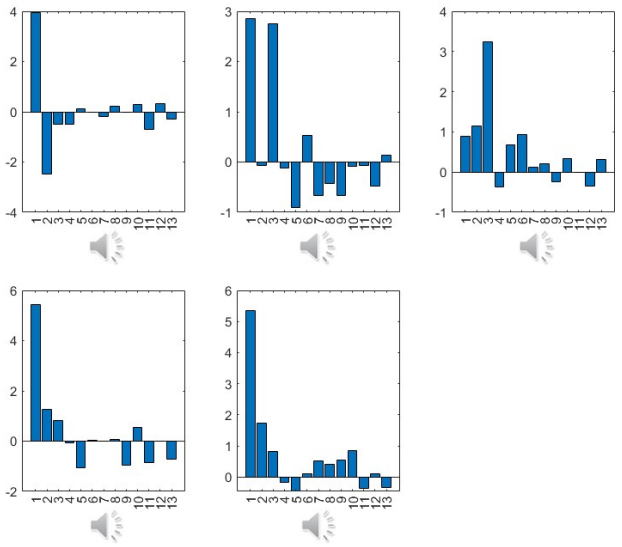
Figure from: https://apexpg.jimdofree.com/matlab-file/mel-frequency-cepstral-coefficients/

56

MFCCs

casa Paganini infomus

- Example:
  13 MFCCs computed on audio samples of the five vowels: a, e, i, o, and u.



57

GeMAPS

casa Paganini infomus

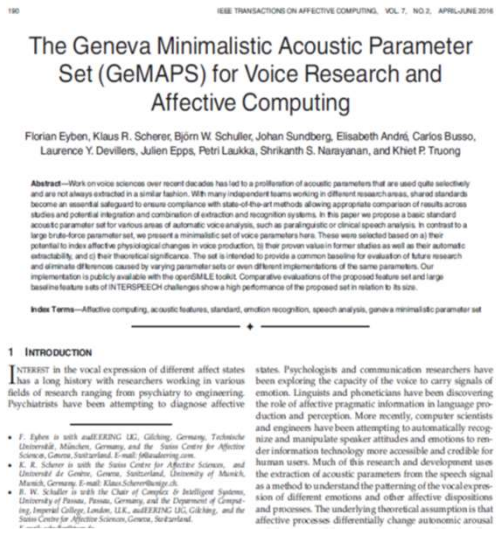
- The Geneva Minimalistic Acoustic Parameter Set (GeMAPS) was conceived at an interdisciplinary meeting of voice and speech scientists in Geneva in 2013 and further developed at Technische Universität München (TUM).
- The choice of parameters was guided by three criteria:
  - The potential of an acoustic parameter to index physiological changes in voice production during affective processes.
  - The frequency and success with which the parameter has been used in the past literature.
  - Its theoretical significance.

58

GeMAPS

casa Paganini infomus

- Two versions of the acoustic parameter set recommendation were proposed: a minimalistic set and an extended set.
- The minimalistic set consists of 18 low-level descriptors divided into 3 groups:
  - Frequency related parameters.
  - Energy/Amplitude related parameters.
  - Spectral (balance) parameters.



190 IEEE TRANSACTIONS ON AFFECTIVE COMPUTING, VOL. 7, NO. 2, APRIL-JUNE 2016

The Geneva Minimalistic Acoustic Parameter Set (GeMAPS) for Voice Research and Affective Computing

Florian Eyben, Klaus R. Scherer, Björn W. Schuller, Johan Sundberg, Elisabeth André, Carlos Busso, Laurence Y. Devillers, Julien Epps, Petri Laukka, Shrikanth S. Narayanan, and Khiet P. Truong

Abstract—Work on voice sciences over recent decades has led to a proliferation of acoustic parameters that are used quite selectively and are not always extracted in a similar fashion. With many independent teams working in different research areas, shared standards become an essential safeguard to ensure compliance with state-of-the-art methods allowing appropriate comparison of results across studies and potential integration and combination of extraction and recognition systems. In this paper we propose a basic standard acoustic parameter set for various areas of automatic voice analysis, such as paralinguistic or clinical speech analysis. In contrast to a large brute-force parameter set, we present a minimalistic set of voice parameters here. These were selected based on a) their potential to index affective physiological changes in voice production, b) their proven value in former studies as well as their automatic extractability, and c) their theoretical significance. The set is intended to provide a common baseline for evaluation of future research and eliminate differences caused by varying parameter sets or even different implementations of the same parameters. Our implementation is publicly available with the openSMILE toolkit. Comparative evaluations of the proposed feature set and large baseline feature sets of INTERSPEECH challenges show a high performance of the proposed set in relation to its size.

Index Terms—Affective computing, acoustic features, standard, emotion recognition, speech analysis, geneva minimalistic parameter set

1 INTRODUCTION

INTEREST in the vocal expression of different affect states has a long history with researchers working in various fields of research ranging from psychiatry to engineering. Psychiatrists have been attempting to diagnose affective states. Psychologists and communication researchers have been exploring the capacity of the voice to carry signals of emotion. Linguists and phoneticians have been discovering the role of affective pragmatic information in language production and perception. More recently, computer scientists and engineers have been attempting to automatically recognize and manipulate speaker attitudes and emotions to render information technology more accessible and credible for human users. Much of this research and development uses the extraction of acoustic parameters from the speech signal as a method to understand the patterning of the vocal expression of different emotions and other affective dispositions and processes. The underlying theoretical assumption is that affective processes differentially change autonomic arousal

- F. Eyben is with audEERING UG, Gilching, Germany, Technische Universität, München, Germany, and the Swiss Centre for Affective Sciences, Geneva, Switzerland. E-mail: f@audeering.com.
- K. R. Scherer is with the Swiss Centre for Affective Sciences, and Université de Genève, Geneva, Switzerland, University of Munich, Munich, Germany. E-mail: Klaus.Scherer@unige.ch.
- B. W. Schuller is with the Chair of Complex & Intelligent Systems, University of Passau, Passau, Germany, and the Department of Computing, Imperial College, London, U.K., audEERING UG, Gilching, and the Swiss Centre for Affective Sciences, Geneva, Switzerland.

59

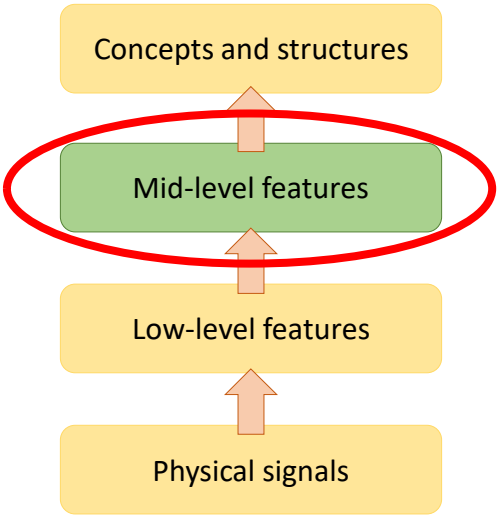
Extended GeMAPS

casa Paganini infomus

- The minimalistic set does not contain any cepstral parameter and only very few dynamic parameters.
- Thus, an extension set to the minimalistic set was proposed which contains 7 further low-level descriptors.
- These consist of 2 frequency related parameters:
  - Formant 2-3 bandwidth: added for completeness to bandwidth of F1, which was already available in the minimalistic set.
- And 5 spectral (balance/shape/dynamics) parameters:
  - MFCC 1-4: Mel-Frequency Cepstral Coefficients 1-4.
  - Spectral flux

60

A conceptual framework



61

Automatic speech segmentation

- Automatic speech segmentation is the partitioning of a speech signal into discrete units (Scharenborg et al., 2010).
- Voice activity detection (VAD), also known as speech activity detection or speech detection, is the detection of the presence or absence of human speech. This is the most basic unitizing process in speech segmentation.
- Other options are, for example, phone boundary detection for isolating single phones and word boundary detection for unitizing single words.

62

Voice activity detection

casa Paganini infomus

- Voice activity detection is usually formalized as a binary classification problem, where the decision concerns the presence of speech for each frame of the input signal.
- Most of the algorithms proposed for VAD can be divided into two processing stages:
  1. Features are extracted from the input speech signal to achieve a representation that discriminates between speech and noise.
  2. A detection scheme is applied to the features resulting in the final decision.

63

Voice activity detection

casa Paganini infomus

- Algorithms use different approaches:
  - Algorithms exploiting general signal characteristics, e.g., energy, statistical descriptors of the content of the signal.
  - Algorithms exploiting one speech-specific feature.
  - Algorithms exploiting multiple signal or speech-specific features.
  - Machine learning approaches.
  - Algorithms fusing information from other sensors/modalities.

64

Energy thresholding

casa Paganini infomus

- This is the simplest approach and consists of applying a selected threshold to the energy of the audio signal.
- Energy is computed over an audio frame:

$$E_n = \sum_{i=n-N+1}^{n} s_i^2$$

$$VAD_n = \begin{cases} 0 & \text{if } E_n < T \\ 1 & \text{if } E_n \geq T \end{cases}$$

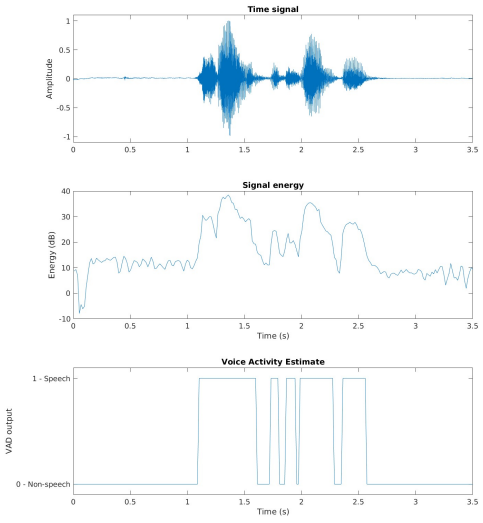
being T the selected threshold.

65

Energy thresholding

casa Paganini infomus

- It is not easy to set a good threshold: speech signals are often corrupted by noise that make thresholding difficult.
- One possibility is to use an adaptive threshold, which is updated during prolonged periods of non-speech intervals consisting of stationary noise (Srinivasant and Gersho,1993)



VAD performed by energy thresholding. Source: https://wiki.aalto.fi/pages/viewpage.action?pageId=151500905.

66

Cepstrum

casa Paganini infomus

- Speech and non-speech segments can be distinguished based on the distance of their cepstra from the cepstrum of a prerecorded noise (Haigh and Mason, 1993):

$$d_n = \frac{1}{H}\sum_{h=1}^{H}\left(c_{nh} - c'_h\right)^2$$

$$VAD_n = \begin{cases} 0 & \text{if } d_n < T \\ 1 & \text{if } d_n \geq T \end{cases}$$

where c_{nh} is the h-th bin of the cepstrum at the n-th frame, c'_h is the h-th bin of the noise cepstrum, H is the number of bins in the cepstrum, and T the selected threshold.

67

G.729 Voice Activity Detection

casa Paganini infomus

- This algorithm is included in the ITU-T G.729 standard, concerning audio compression for telephony (e.g., for VoIP). VAD is used to reduce transmission rate during silence.
- Frame length of 10ms is adopted, i.e., a voice activity decision is made every 10ms.
- After an initialization phase, 4 features are computed:
  - Full-band energy
  - Low-band energy
  - A set of Line Spectral Frequencies (LSF): this is a set of coefficients describing the spectral envelope.
  - Zero crossing rate.

68

G.729 Voice Activity Detection

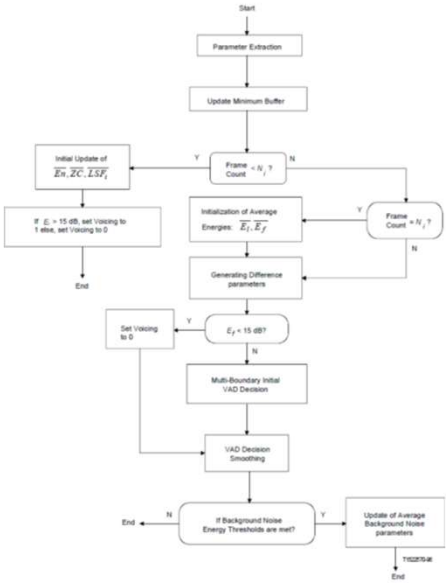
casa Paganini infomus

- Differences are then computed for the 4 features between their values at the current frame and running averages of the values of such features computed over background noise.
- An initial decision is made by applying 14 rules to the difference values. If at least one of such conditions is met, then $VAD_n = 1$, else $VAD_n = 0$.
- Energy considerations, together with neighboring past frames decisions, are used for decision smoothing.
- The running averages are updated only in the presence of background noise. An adaptive threshold is tested, and the update takes place only if the threshold criterion is met.

69

G.729 Voice Activity Detection

casa Paganini infomus



1) if $\Delta S > a_1 \cdot \Delta ZC + b_1$ then $I_{VD} = 1$
2) if $\Delta S > a_2 \cdot \Delta ZC + b_2$ then $I_{VD} = 1$
3) if $\Delta E_f < a_3 \cdot \Delta ZC + b_3$ then $I_{VD} = 1$
4) if $\Delta E_f < a_4 \cdot \Delta ZC + b_4$ then $I_{VD} = 1$
5) if $\Delta E_f < b_5$ then $I_{VD} = 1$
6) if $\Delta E_f < a_6 \cdot \Delta S + b_6$ then $I_{VD} = 1$
7) if $\Delta S > b_7$ then $I_{VD} = 1$
8) if $\Delta E_l < a_8 \cdot \Delta ZC + b_8$ then $I_{VD} = 1$
9) if $\Delta E_l < a_9 \cdot \Delta ZC + b_9$ then $I_{VD} = 1$
10) if $\Delta E_l < b_{10}$ then $I_{VD} = 1$
11) if $\Delta E_l < a_{11} \cdot \Delta S + b_{11}$ then $I_{VD} = 1$
12) if $\Delta E_l > a_{12} \cdot \Delta E_f + b_{12}$ then $I_{VD} = 1$
13) if $\Delta E_l < a_{13} \cdot \Delta E_f + b_{13}$ then $I_{VD} = 1$
14) if $\Delta E_l < a_{14} \cdot \Delta E_f + b_{14}$ then $I_{VD} = 1$

G.729 VAD: flowchart of the algorithm and decision rules. Source: ITU-T G.729 Annex B

70

Other approaches

casa Paganini infomus

- More recent studies address VAD from a machine learning point of view, in which the goal is to classify segments of the input signal into speech and non-speech classes.
- Moreover, multi-class classifiers were also employed to distinguish noise from different kinds of speech, such as plain speech and singing voices (e.g., Kosaka et al., 2018).
- Further, multimodal approaches fusing auditory and visual information were also adopted (e.g., Ariav and Cohen, 2019).

Kosaka, T., Suga, I., and Inoue, M., 2018. Improving Voice Activity Detection for Multimodal Movie Dialogue Corpus. In Proc. *2018 IEEE 7th Global Conference on Consumer Electronics (GCCE)*, 481-484.

Ariav, I., and Cohen, I., 2019. An End-to-End Multimodal Voice Activity Detection Using WaveNet Encoder and Residual Networks. *IEEE Journal of Selected Topics in Signal Processing*, 13, 2, 265-274.

71

Evaluation

casa Paganini infomus

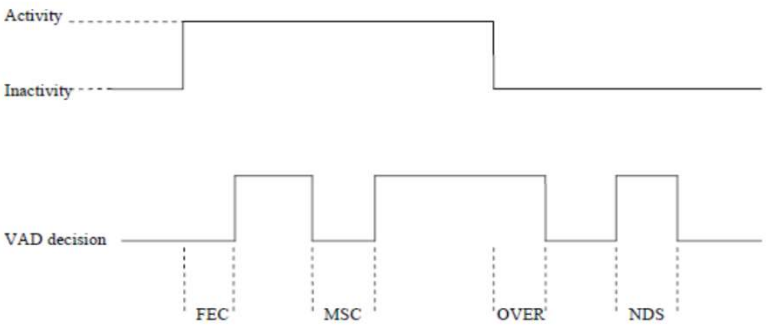
- This is commonly performed on the basis of the following parameters (Beritelli et al., 2002):
  - Front End Clipping (FEC): speech is misclassified as noise in passing from noise to speech activity.
  - Mid Speech Clipping (MSC): speech misclassified as noise during an utterance.
  - Overhang (OVER): noise interpreted as speech in passing from speech activity to noise.
  - Noise Detected as Speech (NDS): noise interpreted as speech within a silence period.

72

Evaluation

casa Paganini infomus

- Evaluation is carried out by comparing the output of a VAD algorithm with a ground-truth created by manual annotation of the presence or absence of voice in a set of recordings.

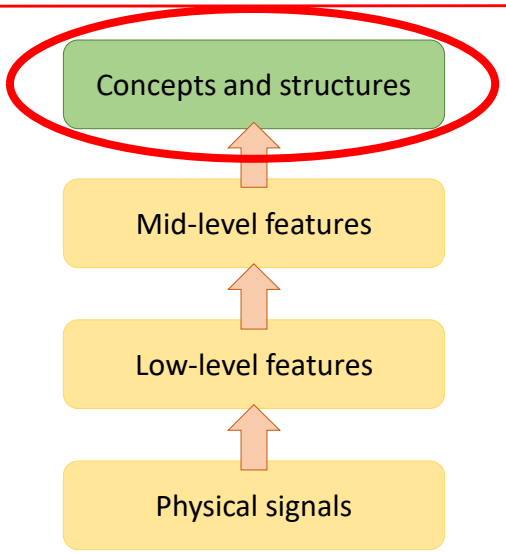


Graphical representation of the parameters for evaluating VAD performance

73

A conceptual framework

casa Paganini infomus

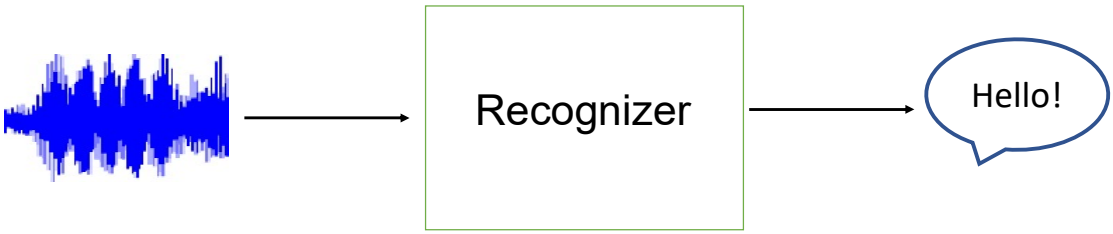


74

Automatic Speech Recognition

casa Paganini infomus

- Task: the task of Automatic Speech Recognition (ASR) is to take as input an acoustic waveform and to produce as output a string of words.



75

Automatic Speech Recognition

casa Paganini infomus

- The first machine recognizing speech was a toy from the 20s (David and Selfridge, 1962).
- Radio Rex was a celluloid dog moving, by means of a spring, when the spring was released by 500 Hz acoustic energy. This is roughly the first formant of vowel [eh] in word *Rex*.
- Rex seemed to come when he was called.

Radio Rex. Source: Jurafsky, D., and Martin, J. H., Speech and Language Processing.

76

Automatic Speech Recognition

casa Paganini infomus

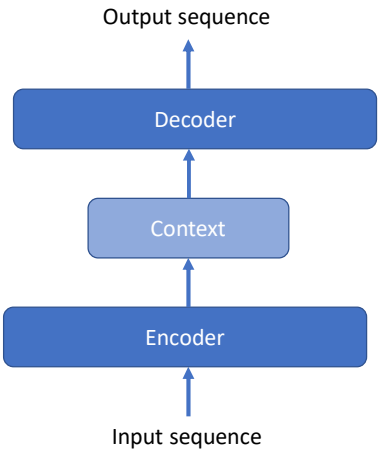
- In the past, Hidden Markov Models (HMMs) have been the most used tool for implementing ASR.
- Nowadays the most common approach to ASR consists of using an architecture called encoder-decoder network.
- This is a sequence-to-sequence network, i.e., a network that takes a sequence as input and predicts a sequence as output.
- Encoder-decoder networks can generate contextually appropriate, arbitrary length, output sequences.

77

Encoder-decoder networks

casa Paganini infomus

- Key idea: an encoder network takes an input sequence and creates a contextualized representation of it (the context); this representation is then passed to a decoder network which generates a task specific output sequence.

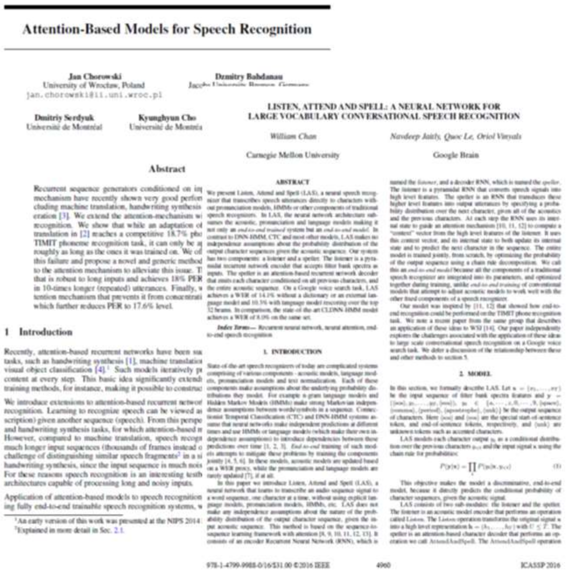


78

Encoder-decoder networks

casa Paganini infomus

- The standard architecture for ASR is referred to as the attention-based encoder decoder (AED) or listen attend and spell (LAS)(Chorowski et al. 2014, Chan et al. 2016).



79

Encoder-decoder networks

casa Paganini infomus

- The input is a sequence of N acoustic feature vectors $\boldsymbol{F} = (\boldsymbol{f}_1, \boldsymbol{f}_2, \dots, \boldsymbol{f}_N)$, one vector per 10ms frame. These are usually vectors of log mel spectral features or MFCCs.
- The output is a sequence of M estimated alphanumerical characters $Y = (\langle SOS \rangle, \hat{y}_1, \hat{y}_2, \dots, \hat{y}_M, \langle EOS \rangle)$, where $\langle SOS \rangle$ and $\langle EOS \rangle$ are special tokens for start of sequence and end of sequence. For example, one can take $\hat{y}_i \in A$ where:

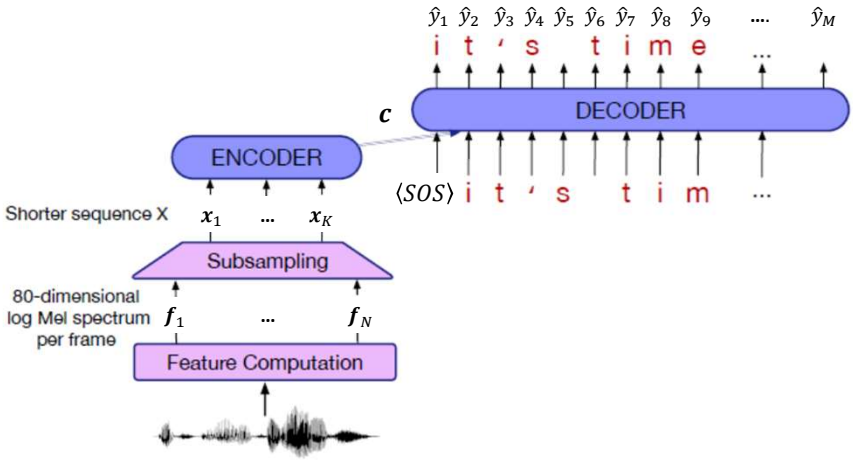
$$A = \begin{Bmatrix} a, b, c, \dots, z, 0,1, \dots 9, \langle space \rangle, \langle comma \rangle, \\ \langle period \rangle, \langle apostroph \rangle, \langle unknown \rangle \end{Bmatrix}$$

80

Encoder-decoder networks

casa Paganini infomus

- Typical encoder-decoder architecture for ASR:



Adapted from Jurafsky, D., and Martin, J. H., Speech and Language Processing.

81

Subsampling

casa Paganini infomus

- Input and output sequences have stark length differences: very long acoustic feature sequences map to much shorter sequences of letters.
- Example: a single word might be 5 letters long, but it may take 200 acoustic frames (supposing the word lasts about 2 seconds and frame duration is 10ms).
- The subsampling module in the architecture shortens the acoustic feature sequence before the encoder stage, so that the input sequence $\boldsymbol{F} = (\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_N)$ is transformed into a shorter sequence $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K)$ being $K < N$.

82

Subsampling

casa Paganini infomus

- The simplest algorithm is a method called low frame rate (Pundak and Sainath, 2016):
  - Acoustic feature vector $\boldsymbol{f}_i$ is concatenated with the prior two vectors $\boldsymbol{f}_{i-1}$ and $\boldsymbol{f}_{i-2}$ to make a new vector $\boldsymbol{x}_j$ three times longer.
  - Vectors $\boldsymbol{f}_{i-1}$ and $\boldsymbol{f}_{i-2}$ are then deleted.
- Example: instead of a 40-dimensional acoustic feature vector every 10ms, we have a 120-dimensional acoustic feature vector every 30ms. The new input vector is 3 times longer, the sequence length is 3 times shorter ($K = N/3$).
- Other more sophisticated approaches (e.g., convolutional networks or pyramidal RNNs) can also be employed.

83

Encoder

casa Paganini infomus

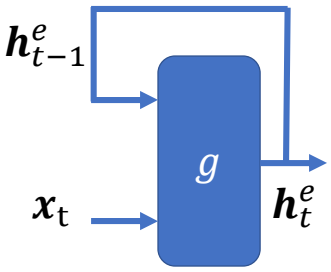
- The encoder accepts an input sequence of subsampled acoustic feature vectors $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_K)$ and generates a corresponding sequence of contextualized representations $\boldsymbol{H} = (\boldsymbol{h}_1^e, \boldsymbol{h}_2^e, \dots, \boldsymbol{h}_K^e)$.
- The essence of sequence $\boldsymbol{H}$ is condensed in the context vector $\boldsymbol{c}$, which is a function of $\boldsymbol{H} = (\boldsymbol{h}_1^e, \boldsymbol{h}_2^e, \dots, \boldsymbol{h}_K^e)$.
- Several options are available for implementing the encoder, ranging from simple RNNs to LSTMs, GRUs, convolutional networks, and transformers.

84

Example: encoder with RNN

casa Paganini infomus

- Current representation $\boldsymbol{h}_t^e$ is a non-linear function of the current input $\boldsymbol{x}_t$, of the previous representation vector $\boldsymbol{h}_{t-1}^e$, and of a set of parameters $\boldsymbol{\theta}$, that is: $\boldsymbol{h}_\mathrm{t} = g(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}^e, \boldsymbol{\theta})$.
- Example: $\boldsymbol{h}_t^e = ReLU(W\boldsymbol{x}_\mathrm{t} + \mathrm{U}\boldsymbol{h}_{t-1}^e)$.
- Parameters $\boldsymbol{\theta}$ (e.g., matrices W and U) are learned from labelled samples collected in a training set. These usually consist of a collection of spoken sentences and their transcript.

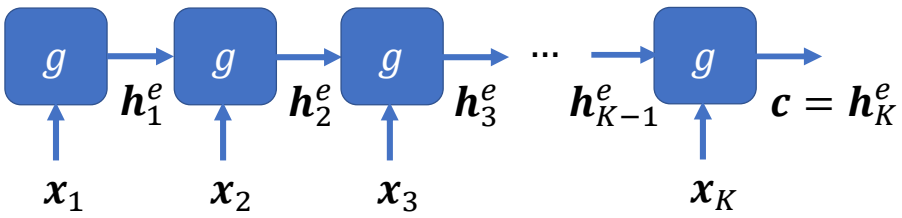


85

Example: encoder with RNN

casa Paganini infomus

- This structure can be unrolled over time.
- The context vector $\boldsymbol{c}$ is the final representation that the encoder produces after processing the last input in the sequence, that is $\boldsymbol{c} = \boldsymbol{h}_K^e$.
- The same context vector $\boldsymbol{c}$ is used for initializing the internal representations of the decoder, i.e., $\boldsymbol{h}_0^d = \boldsymbol{c}$.



86

Decoder

casa Paganini infomus

- The decoder accepts as input the context vector $\boldsymbol{c}$ produced by the encoder, and generates an arbitrary length sequence of hidden states $H^d = \left(\boldsymbol{h}_1^d, \boldsymbol{h}_2^d, \ldots, \boldsymbol{h}_M^d\right)$
- From the sequence of hidden states H^d a sequence of outputs $\hat{Y} = (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_{M-1}, \langle EOS \rangle)$ is obtained.
- As for the encoder, several options are available for implementing the decoder, ranging again from simple RNNs to LSTMs, GRUs, convolutional networks, and transformers.

87

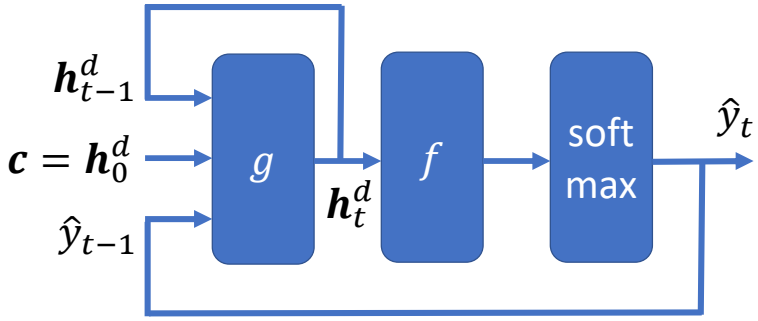
Example: decoder with RNN

casa Paganini infomus

- The first hidden state is a non-linear function of the start of sequence symbol, the context vector $\boldsymbol{h}_0^d = \boldsymbol{c}$, and of a set of parameters $\boldsymbol{\theta}$, that is: $\boldsymbol{h}_1^d = g(\langle SOS \rangle, \boldsymbol{h}_0^d, \mathbf{c}, \boldsymbol{\theta})$.
- Current hidden state $\boldsymbol{h}_t^d$ is a non-linear function of the previously generated output $\hat{y}_{t-1}$, of the previous hidden state $\boldsymbol{h}_{t-1}^d$, of the context vector, and of a set of parameters $\boldsymbol{\theta}$, that is: $\boldsymbol{h}_t^d = g(\hat{y}_{t-1}, \boldsymbol{h}_{t-1}^d, \mathbf{c}, \boldsymbol{\theta})$.
- Output $\hat{y}_t$ is obtained by applying softmax to a function of hidden state $\boldsymbol{h}_t^d$, i.e., $\boldsymbol{s}_t = \text{softmax}(f(\boldsymbol{h}_t^d, \boldsymbol{\rho}))$, and looking for the item of $\boldsymbol{s}_t$ having the maximum value.
- Parameters $\boldsymbol{\theta}$ and $\boldsymbol{\rho}$ are learned during training.

88

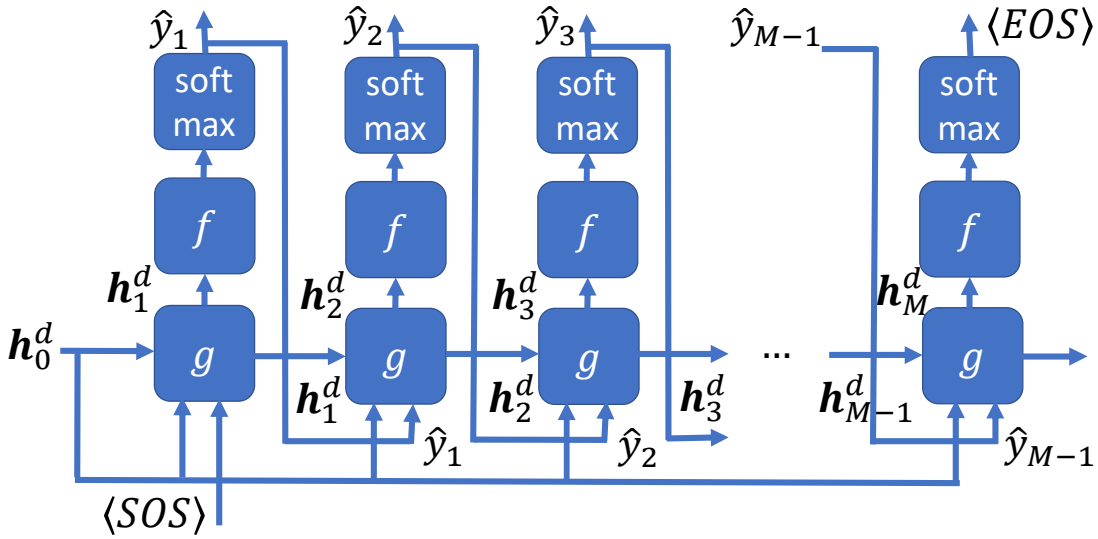
Example: decoder with RNN



- Note: if $\boldsymbol{z}_t = f(\boldsymbol{h}_t^d, \boldsymbol{\rho}) \in \mathbb{R}^{|A|}$, $\boldsymbol{s}_t = \text{softmax}(\boldsymbol{z}_t) = [s_{t1}, \dots, s_{t|A|}]$, where $s_{ti} = \exp(z_{ti}) / \sum_{i=1}^{|A|} \exp(z_{ti})$. The selected output $\hat{y}_t$ is the character in correspondence of $i_t^* = \underset{i=1\dots|A|}{\text{argmax}}\, s_{ti}$.
- This structure can also be unrolled over time.

89

Example: decoder with RNN



90

ASR evaluation

casa Paganini infomus

- The standard evaluation metric for automatic speech recognition systems is the word error rate (WER).
- This is the extent to which the word string returned by the recognizer (that is the hypothesized string) differs from a reference transcription.
- The first step consists of computing the minimum edit distance in words between hypothesized and correct strings.
- This is the minimum number of word substitutions, word insertions, and word deletions necessary to map between the correct and hypothesized strings.

91

ASR evaluation

casa Paganini infomus

- The word error rate is then defined as follows:

$$\text{WER} = 100 \cdot \frac{n_i + n_s + n_d}{n}$$

where n_i is the number of insertions, n_s the number of substitutions, n_d the number of deletions, and n is total number of words in the correct transcript.

- Example: WER = (3+6+1)/13 = 76.9%.

REF:	i	***	**	UM	the	PHONE	IS	i	LEFT	THE	portable	****	PHONE	UPSTAIRS	last	night
HYP:	i	GOT	IT	TO	the	*****	FULLEST	i	LOVE	TO	portable	FORM	OF	STORES	last	night
Eval:		I	I	S		D	S		S	S		I	S	S		

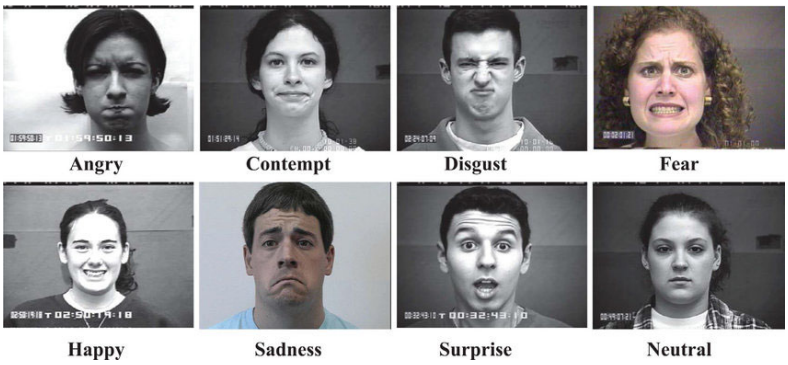
92

4. Facial expression

1

Facial expression

- A facial expression is one or more motions or positions of the muscles beneath the skin of the face (Freitas-Magalhães, 2011).
- Facial expressions are a form of nonverbal communication. They are a primary means of conveying social information between humans.

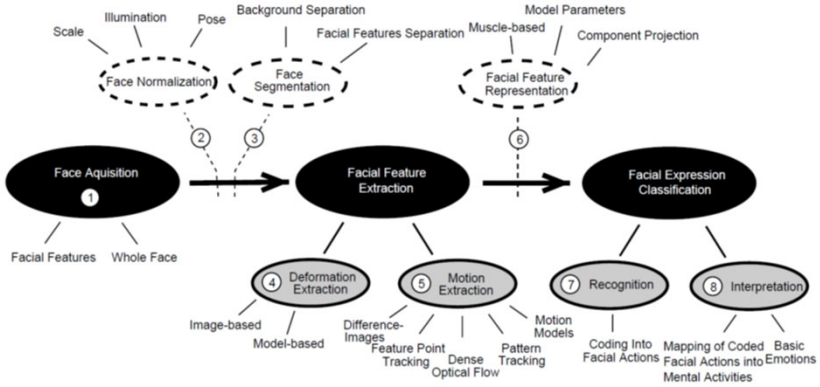


2

Analysis framework

casa Paganini infomus

- Fasel and Luettin (2003) proposed an analysis framework for facial expression that does not differ significantly from the general layered framework for multimodal systems.



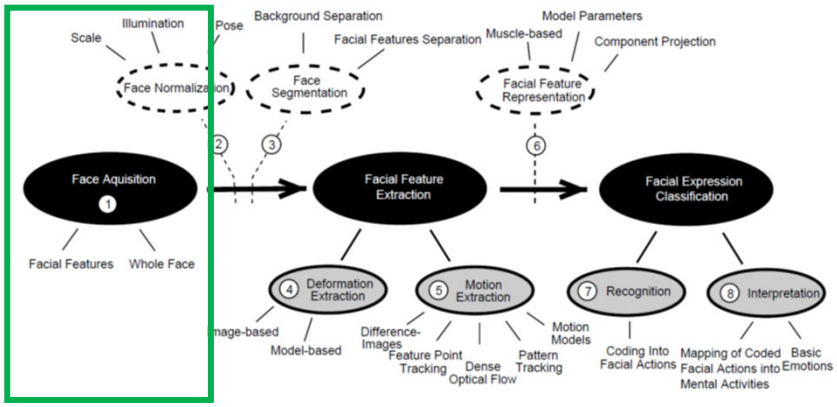
Generic facial expression analysis framework (Fasel and Luettin, 2003)

3

Analysis framework

casa Paganini infomus

- Face acquisition: this step concerns locating and tracking faces in complex scenes with cluttered backgrounds.

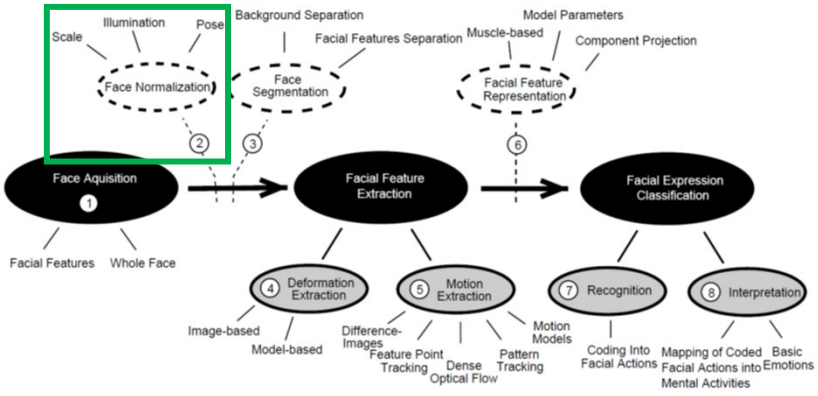


4

Analysis framework

casa Paganini infomus

- Face normalization: appearance changes may be due to variations of pose or illumination. It is a good practice to normalize faces with respect to such sources of variation.

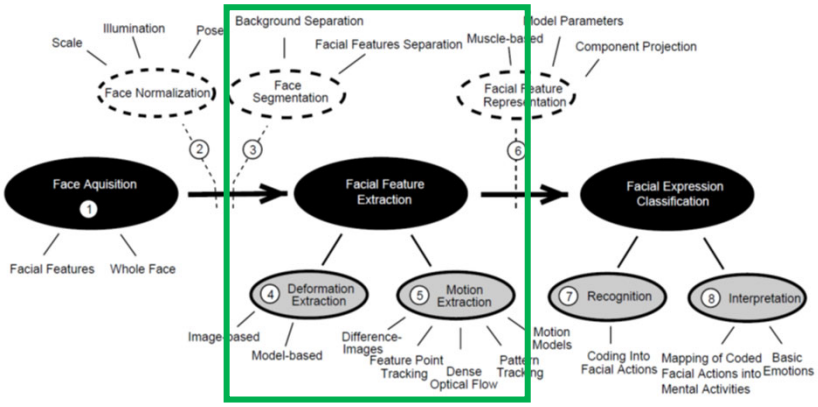


5

Analysis framework

casa Paganini infomus

- Facial feature extraction: features may focus on motion or deformation of faces. They may act locally or holistically.

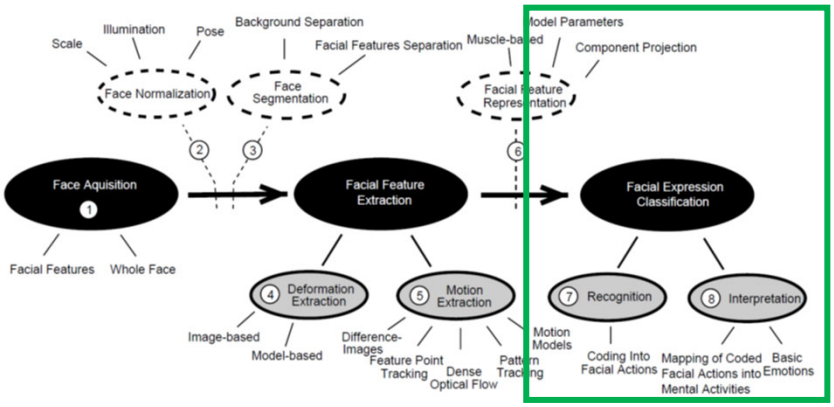


6

Analysis framework

casa Paganini infomus

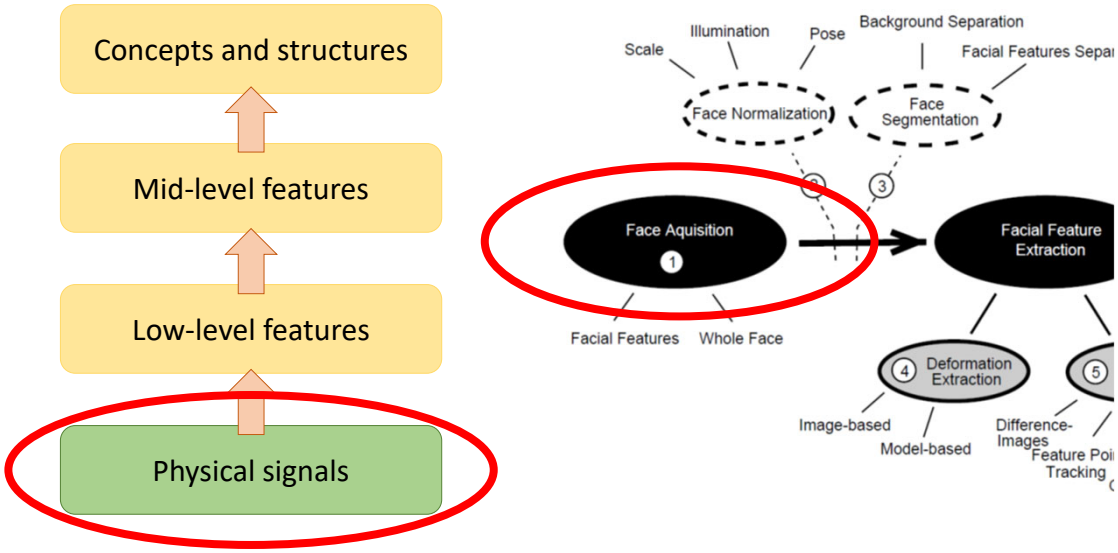
- Facial expression classification: this concerns unitizing, annotation, and classification (e.g., by means of ML).



7

A conceptual framework

casa Paganini infomus



8

Devices

casa Paganini infomus

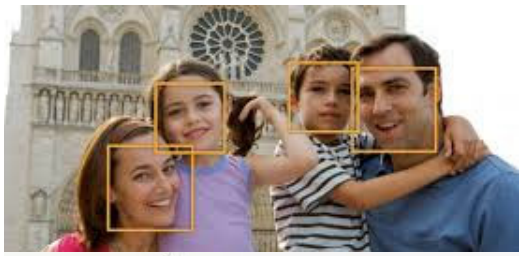
- These are usually the same video-based systems used for capturing movement.
- Of course, devices and possible markers may need to be set-up differently.

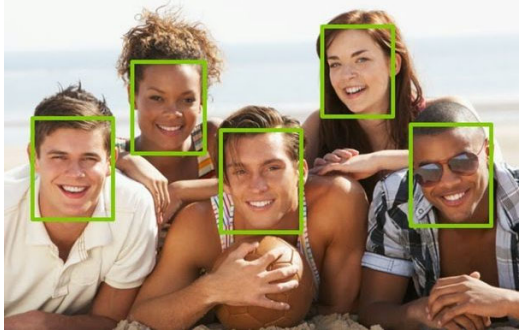
9

Face detection

casa Paganini infomus

- It is one of the first computer vision applications and the first step for face analysis.
- It can be regarded as a special case of object-class detection.
- The problem consists of detecting and localizing human faces within an image:
  - Input: an image.
  - Output: bounding boxes of the detected faces.





10

Face detection

casa Paganini infomus

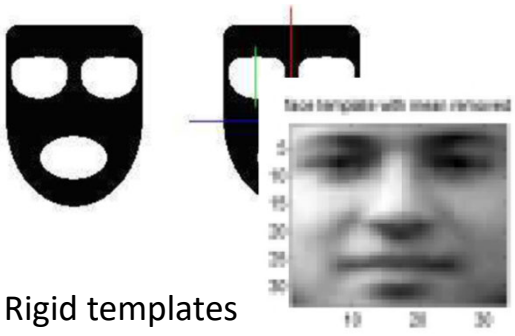
- A classification of face detection algorithms (Yang, Kriegman, and Ahuja, 2002):
  - Knowledge-based methods: use pre-defined rules based on human knowledge to detect a face.
  - Feature invariant methods: find face structure features robust to pose and lighting variations.
  - Template matching methods: use pre-stored face templates to determine where a human face is depicted in an image.
  - Appearance-based methods: learn face models from a set of representative training face images.

11

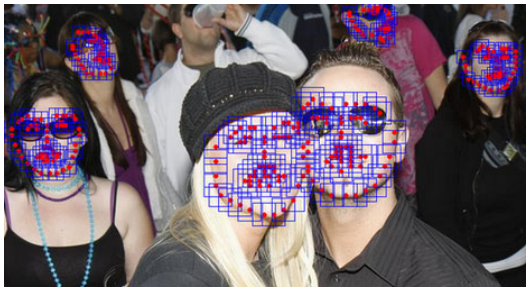
Face detection

casa Paganini infomus

- A more recent classification (Zafeiriou et al., 2015):
  - Algorithms based on rigid templates.
  - Algorithms applying Deformable Parts-based Models (DPM).



Rigid templates



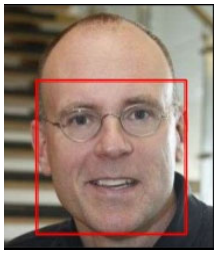
Deformable Parts-based Models

12

Viola-Jones face detector

casa Paganini infomus

- Viola and Jones (2001, 2004) face detector was historically the first algorithm making face detection practically feasible in real-world applications.
- Until today, it is widely applied in digital cameras and photo organization software.



Paul Viola

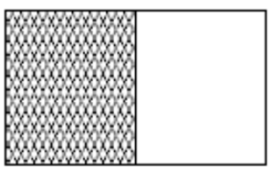
Michael J. Jones

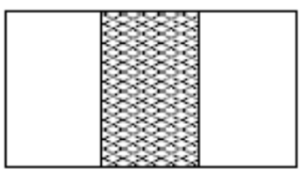
13

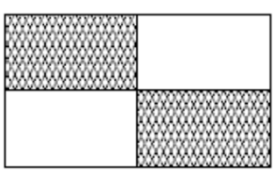
Viola-Jones face detector

casa Paganini infomus

- This face detector classifies images based on the value of three kinds of features:
  - Two-rectangle feature: its value is the difference between the sum of the pixel values within two adjacent rectangular regions.
  - Three-rectangle feature: its value is the sum of the pixels values within two outside rectangles subtracted from the sum of the pixel values in a center rectangle.
  - Four-rectangle feature: its value is the difference between the sum of the pixel values within two diagonal pairs of rectangles.

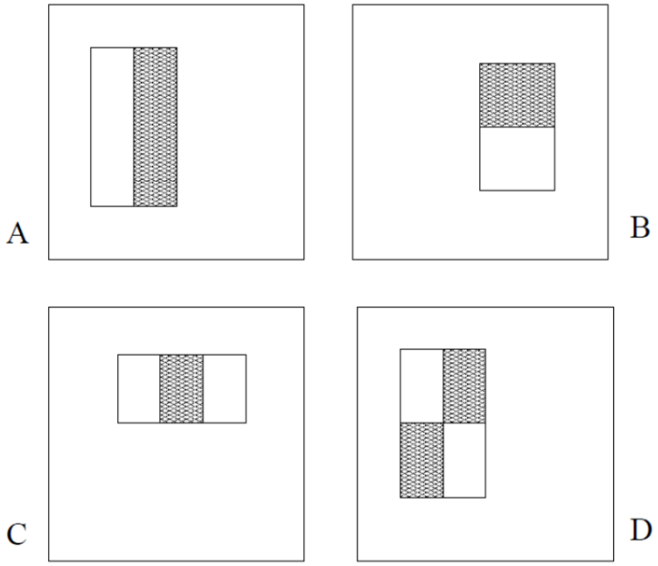






14

Viola-Jones face detector



Example of rectangle features: the sum of the pixel values lying within the white rectangles are subtracted from the sum of pixels values in the grey rectangles. Picture from Viola and Jones (2001).

15

Viola-Jones face detector

- Rectangle features can be computed very rapidly using an intermediate representation called integral image.
- The integral image at location (*x*, *y*) contains the sum of the pixel values above and to the left of *x* and *y*, inclusive:

$$ii(x,y) = \sum_{x' \leq x, y' \leq y} i(x',y').$$

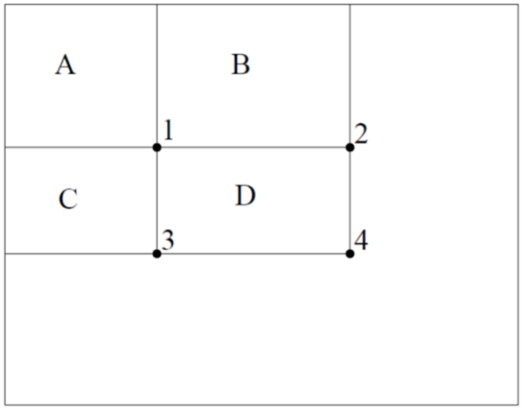
- The integral image is efficient to compute: it can be computed in one pass over the original image.

16

Viola-Jones face detector

- Any rectangular sum is computed in four array references.
- The difference between two rectangular sums needs eight references.

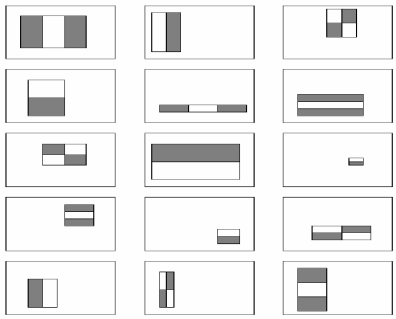
Example: The value of the integral image at location 1 is the sum of the pixels in rectangle A. The value at location 2 is A+B, at location 3 is A+C, and at location 4 is A+B+C+D. The sum within D is computed as 4+1 - (2+3).



17

Viola-Jones face detector

- Viola-Jones face detector uses a 24x24 sub-window.
- In such a sub-window there are ~160.000 features: too many! We cannot compute all of them during the detection process.
- A subset of relevant features has to be selected



Relevant feature

Irrelevant feature

Picture by Kostantina Palla and Alfredo Kalaitzis

18

Viola-Jones face detector

casa Paganini infomus

- Classification is performed using the Adaptive Boost (AdaBoost) technique, over the most relevant features.
- This is an iterative algorithm that builds a "strong" classifier as a linear combination of weighted "weak" classifiers.
- Each weak classifier uses just one feature.

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$$ Weak classifier

$$h(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise} \end{cases}$$ Strong classifier

19

Viola-Jones face detector

casa Paganini infomus

Given:
N images labeled + (face) or - (no face).
Images are given weights w.
Initially, all weights w are set equally.
Repeat T times:
Step 1:
Choose the most efficient weak classifier. Threshold θ is estimated to maximize accuracy. Assign the classifier a weight α proportional to its accuracy.
Step 2:
Update the weights w to emphasize the examples which were incorrectly classified. This makes the next weak classifier to focus on "harder" samples.
Result:
The final (strong) classifier is a weighted combination of the T "weak" classifiers, weighted according to their accuracy.

20

Viola-Jones face detector

casa Paganini infomus

- A 200-feature classifier achieved:
  - 95% detection rate
  - 0.14×10^{-3} false positive rate (1 in 14084)
  - Scans all sub-windows of a 384x288 pixel image in 0.7 seconds (on Intel PIII 700MHz)
- Verdict:
  good and fast, but not enough!

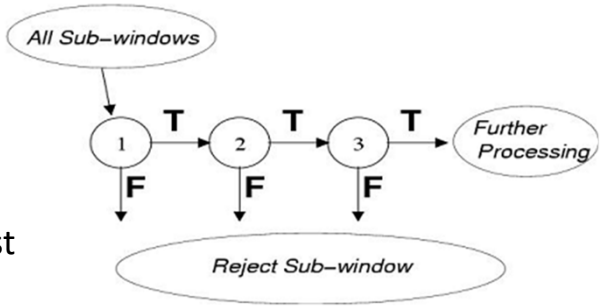
First and second feature selected by AdaBoost (Viola and Jones, 2001)

21

Viola-Jones face detector

casa Paganini infomus

- To improve performance, we notice that:
  - On average only 0.01% of sub-windows are faces.
  - But equal time is spent on all sub-windows.
  - Then, we should spend most time only on potentially positive sub-windows.
- Solution:
  a cascade of classifiers.



A cascade of three classifiers

22

Viola-Jones face detector

casa Paganini infomus

- A simple 2-feature classifier can achieve almost 100% detection rate with 50% false positive rate.
- That classifier can act as the first layer of a series of classifiers to filter out most negative windows.
- The second layer with 10 features can tackle “harder” negative-windows, which survived the first layer, and so on…
- A cascade of gradually more complex classifiers achieves better detection rates.
- The final Viola-Jones face detector is a 38-layer cascade of classifiers, including a total of 6060 features.

23

Viola-Jones face detector

casa Paganini infomus

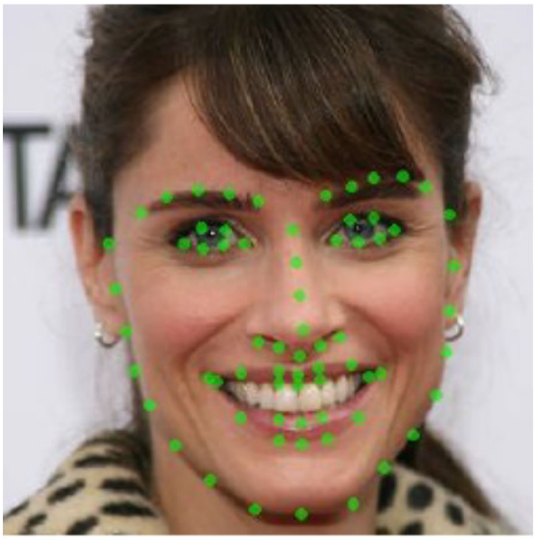
- Some figures:
  - Training time was on the order of weeks!
  - Since a large majority of the sub-windows are discarded by the first two stages of the cascade, an average of 8 features out of a total of 6060 are evaluated per sub-window.
  - On a 700 Mhz Pentium III processor, the face detector can process a 384 by 288-pixel image in ~0.067s (i.e., ~15fps).
  - Detection rate on the MIT test set was 77.8% with 5 false positives.

Pictures in (Viola and Jones, 2004).

24

Facial landmark localization

- Facial landmarks are defined as distinctive face locations, such as corners of the eyes, center of the bottom lip, tip of the nose, and so on.
- Taken in sufficient numbers they define the face shape.
- Localization and tracking of facial landmarks can improve accuracy of analysis of facial expression.



25

Supervised Descent Method

- A quite recent and common approach to facial landmark localization is the Supervised Descent Method (SDM), proposed by Xiong and De la Torre in 2013.
- SDM minimizes a Non-linear Least Squares (NLS) function.
- During training, the SDM learns a sequence of descent directions that minimizes the mean of NLS functions sampled at different points.
- In testing, SDM minimizes the NLS objective using the learned descent directions.

26

Newton's method

casa Paganini infomus

- The method takes inspiration from Newton's method to find the minimum of a nonlinear function $f: \mathbb{R}^{n\times 1} \longmapsto \mathbb{R}$.
- If $f(\boldsymbol{x}) \in \mathcal{C}^2$ and can be approximated by a quadratic function in a neighborhood of the minimum, the method finds a sequence $\{\boldsymbol{x}_k\}_{k\in\mathbb{N}}$ converging to the minimum:

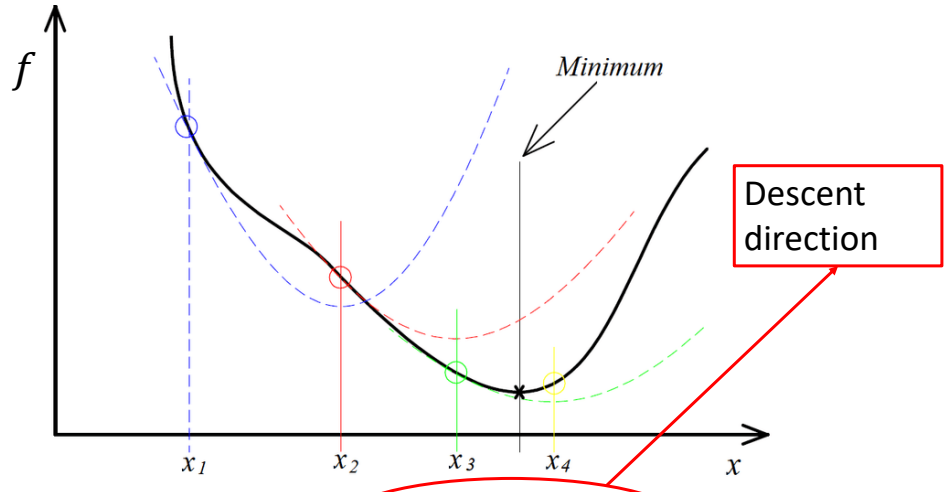
$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \mathrm{H}_f^{-1}(\boldsymbol{x}_k) \cdot \nabla f(\boldsymbol{x}_k)$$

being $\mathrm{H}^{-1}(\boldsymbol{x}_k)$ $(\in \mathbb{R}^{n\times n})$ and $\nabla f(\boldsymbol{x}_k)$ $(\in \mathbb{R}^{n\times 1})$ the inverse of the Hessian matrix and the gradient of f calculated in $\boldsymbol{x}_k$.

27

Newton's method

casa Paganini infomus

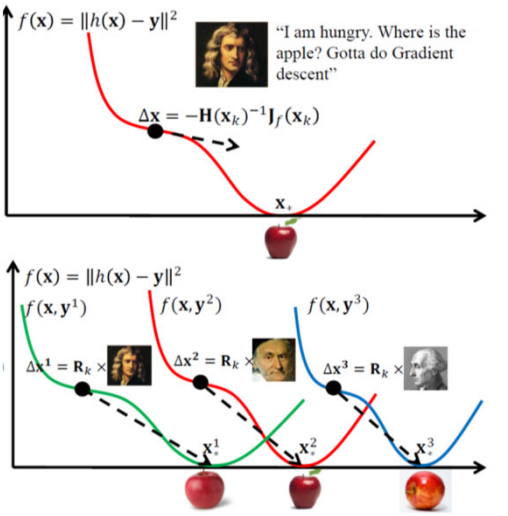


$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \mathrm{H}_f^{-1}(\boldsymbol{x}_k) \cdot \nabla f(\boldsymbol{x}_k)$$

28

Supervised Descent Method

- Concept of SDM: remove the constraints about smoothness of function f and properties of matrix H (required to be definite positive) by learning descent directions from data.
- This also improves computational complexity, because we do not have to compute $\mathrm{H}_f^{-1}(\boldsymbol{x}_k)$.



Pictures taken from (Xiong and De la Torre, 2013).

29

SDM: problem formulation

- Let us take an image $\boldsymbol{d} \in \mathbb{R}^{m\times 1}$ (i.e., having *m* pixels).
- $\boldsymbol{x} \in \mathbb{R}^{2p\times 1}$ are the coordinates of *p* landmarks.
- Faces in the training set are manually annotated and, for each of them, landmark coordinates are $\boldsymbol{x}_* \in \mathbb{R}^{2p\times 1}$.
- The algorithm is initialized by assigning landmarks the initial coordinates $\boldsymbol{x}_0 \in \mathbb{R}^{2p\times 1}$.
- $\boldsymbol{d}(\boldsymbol{x}) \in \mathbb{R}^{p\times 1}$ are the values of the landmark pixels.
- $\boldsymbol{h}(\boldsymbol{x}) \in \mathbb{R}^{kp\times 1}$ is a vector of features describing the neighborhood of each landmark (*k* values per landmark).

30

SDM: problem formulation

- The problem of landmark localization can be formulated as finding the displacement $\Delta \boldsymbol{x}$ with respect to the initial coordinates $\boldsymbol{x}_0$ that minimizes the following function:

$$f(\boldsymbol{x}_0 + \Delta \boldsymbol{x}) = \|\boldsymbol{h}(\boldsymbol{x}_0 + \Delta \boldsymbol{x}) - \boldsymbol{h}(\boldsymbol{x}_*)\|^2$$

- That is, we are looking for:

$$\min_{\Delta \boldsymbol{x}} f(\boldsymbol{x}_0 + \Delta \boldsymbol{x}) = \min_{\Delta \boldsymbol{x}} \|\boldsymbol{h}(\boldsymbol{x}_0 + \Delta \boldsymbol{x}) - \boldsymbol{h}(\boldsymbol{x}_*)\|^2$$

31

Supervised Descent Method

- The Newton's method can find a sequence of displacements $\{\Delta \boldsymbol{x}_k\}_{k \in \mathbb{N}}$ converging to the minimum of f.
- Such sequence can be proven to be:

$$\Delta \boldsymbol{x}_{k+1} = \Delta \boldsymbol{x}_k - 2\mathrm{H}_f^{-1}(\boldsymbol{x}_0 + \Delta \boldsymbol{x}_k)\, J_{\boldsymbol{h}}^T(\boldsymbol{x}_0 + \Delta \boldsymbol{x}_k)\big(\boldsymbol{h}(\boldsymbol{x}_0 + \Delta \boldsymbol{x}_k) - \boldsymbol{h}(\boldsymbol{x}_*)\big)$$

being:

- $\mathrm{H}_f{}^{-1}(\boldsymbol{x}_0 + \Delta \boldsymbol{x}_k) \in \mathbb{R}^{2p \times 2p}$ the inverse of the Hessian matrix of f calculated in $\boldsymbol{x}_0 + \Delta \boldsymbol{x}_k$.
- $J_{\boldsymbol{h}}^T(\boldsymbol{x}_0 + \Delta \boldsymbol{x}_k) \in \mathbb{R}^{2p \times kp}$ the transposed Jacobian matrix of $\boldsymbol{h}$ calculated in $\boldsymbol{x}_0 + \Delta \boldsymbol{x}_k$.

32

Supervised Descent Method

casa Paganini infomus

- This can be rewritten as: $\Delta \boldsymbol{x}_{k+1} = \Delta \boldsymbol{x}_k + \mathrm{R}_k \Delta \boldsymbol{\phi}_k$ where:
  - $\mathrm{R}_k = -2\mathrm{H}_f^{-1}(\boldsymbol{x}_0 + \Delta \boldsymbol{x}_k)\, J_{\boldsymbol{h}}^T(\boldsymbol{x}_0 + \Delta \boldsymbol{x}_k)$
  - $\boldsymbol{\phi}_k = \boldsymbol{h}(\boldsymbol{x}_0 + \Delta \boldsymbol{x}_k)$, $\boldsymbol{\phi}_* = \boldsymbol{h}(\boldsymbol{x}_*)$
  - $\Delta \boldsymbol{\phi}_k = \boldsymbol{\phi}_k - \boldsymbol{\phi}_* = \boldsymbol{h}(\boldsymbol{x}_0 + \Delta \boldsymbol{x}_k) - \boldsymbol{h}(\boldsymbol{x}_*)$
- Since during testing the algorithm does not use the training information $\boldsymbol{\phi}_*$, the equation can be finally reformulated as:

$$\Delta \boldsymbol{x}_{k+1} = \Delta \boldsymbol{x}_k + \mathrm{R}_k \boldsymbol{\phi}_k + \boldsymbol{b}_k$$

  being $\boldsymbol{b}_k$ a bias term.
- SDM learns matrices R_k and bias terms $\boldsymbol{b}_k$.

33

SDM: training

casa Paganini infomus

- The dataset for training consists of N faces $\boldsymbol{d}^i, i = 1 \ldots N$ and the manually annotated positions of their landmarks $\boldsymbol{x}_*^i$.
- For each face, by applying the above-mentioned equation, the first descent direction is $\Delta \boldsymbol{x}_0^i = \mathrm{R}_0 \boldsymbol{\phi}_0^i + \boldsymbol{b}_0$.
- Thus, the first training step consists of learning R_0 and $\boldsymbol{b}_0$.
- This is obtained by:
  - Randomly assigning initial positions $\boldsymbol{x}_0^i$.
  - Computing the new position $\boldsymbol{x}_1^i$, which is reached by following the first descent direction $\Delta \boldsymbol{x}_0^i$.
  - Minimizing the error, i.e., the distance from the target $\boldsymbol{x}_*^i$.

34

SDM: training

casa Paganini infomus

- More formally, being $\boldsymbol{X}_0^i$ a random vector expressing the randomness of the initial position $\boldsymbol{x}_0^i$:
  - $\Delta\boldsymbol{X}_0^i = \boldsymbol{X}_1^i - \boldsymbol{X}_0^i = \mathrm{R}_0\boldsymbol{\Phi}_0^i + \boldsymbol{b}_0$, being $\boldsymbol{\Phi}_0^i = \boldsymbol{h}(\boldsymbol{X}_0^i)$.
  - The new position which is reached is $\boldsymbol{X}_1^i = \boldsymbol{X}_0^i + \mathrm{R}_0\boldsymbol{\Phi}_0^i + \boldsymbol{b}_0$
  - The error $e(\boldsymbol{X}_0^i)$ we make is:

$$e(\boldsymbol{X}_0^i) = \left\|\boldsymbol{x}_*^i - \boldsymbol{X}_1^i\right\|^2 = \left\|\boldsymbol{x}_*^i - (\boldsymbol{X}_0^i + \mathrm{R}_0\boldsymbol{\Phi}_0^i + \boldsymbol{b}_0)\right\|^2 =$$
$$= \left\|\boldsymbol{x}_*^i - \boldsymbol{X}_0^i - \mathrm{R}_0\boldsymbol{\Phi}_0^i - \boldsymbol{b}_0\right\|^2 = \left\|\Delta\boldsymbol{X}_{*0}^i - \mathrm{R}_0\boldsymbol{\Phi}_0^i - \boldsymbol{b}_0\right\|^2$$

being $\Delta\boldsymbol{X}_{*0}^i = \boldsymbol{x}_*^i - \boldsymbol{X}_0^i$.

35

SDM: training

casa Paganini infomus

- Assuming that random vector $\boldsymbol{X}_0^i$ has a probability density function $f_{\boldsymbol{X}_0^i}(\boldsymbol{x}_0^i)$, we then compute:

$$\arg\min_{\mathrm{R}_0,\,\boldsymbol{b}_0} \sum_{i=1}^{N} \mathbb{E}\left(e(\boldsymbol{X}_0^i)\right) = \arg\min_{\mathrm{R}_0,\,\boldsymbol{b}_0} \sum_{i=1}^{N} \int e(\boldsymbol{x}_0^i) f_{\boldsymbol{X}_0^i}(\boldsymbol{x}_0^i) d\boldsymbol{x}_0^i$$

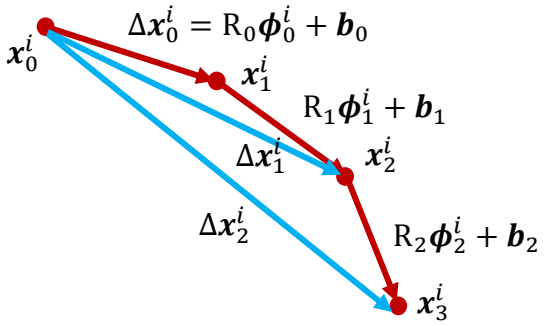
- Supposing now that $f_{\boldsymbol{X}_0^i}(\boldsymbol{x}_0^i)$ is Gaussian, the integral is approximated by means of Montecarlo sampling, yielding:

$$\arg\min_{\mathrm{R}_0,\,\mathbf{b}_0} \sum_{i=1}^{N} \sum_{\boldsymbol{x}_0^i} \left\|\Delta\boldsymbol{x}_{*0}^i - \mathrm{R}_0\boldsymbol{\phi}_0^i - \boldsymbol{b}_0\right\|^2$$

36

SDM: training

- Such minimization is the well-known linear least squares problem, which can be solved in closed-form.
- After R_0 and $\boldsymbol{b}_0$ are learnt, we move to $\boldsymbol{x}_1^i$ and apply the same approach to learn R_1 and $\boldsymbol{b}_1$.



37

SDM: training

- More generally, at step k we move to $\boldsymbol{x}_k$ and we learn R_k and $\boldsymbol{b}_k$. Then:
  - We can now get the next increment: $\Delta \boldsymbol{x}_{k+1}^i = \Delta \boldsymbol{x}_k^i + \mathrm{R}_k \boldsymbol{\phi}_k^i + \boldsymbol{b}_k$.
  - This enables us to move to $\boldsymbol{x}_{k+1}^i$. We recall indeed that our initial problem was to minimize $f(\boldsymbol{x}_0 + \Delta \boldsymbol{x})$, so $\boldsymbol{x}_k = \boldsymbol{x}_0 + \Delta \boldsymbol{x}_k$. That is: $\Delta \boldsymbol{x}_{k+1}^i = \boldsymbol{x}_{k+1}^i - \boldsymbol{x}_0^i$ and $\Delta \boldsymbol{x}_k^i = \boldsymbol{x}_k^i - \boldsymbol{x}_0^i$.
  - Thus, $\Delta \boldsymbol{x}_{k+1}^i = \boldsymbol{x}_{k+1}^i - \boldsymbol{x}_0^i = \Delta \boldsymbol{x}_k^i + \mathrm{R}_k \boldsymbol{\phi}_k^i + \boldsymbol{b}_k = \boldsymbol{x}_k^i - \boldsymbol{x}_0^i + \mathrm{R}_k \boldsymbol{\phi}_k^i + \boldsymbol{b}_k$. So that: $\boldsymbol{x}_{k+1}^i = \boldsymbol{x}_k^i + \mathrm{R}_k \boldsymbol{\phi}_k^i + \boldsymbol{b}_k$.
  - Being initial positions randomly assigned, $\boldsymbol{X}_0^i$ is indeed a random vector and so are $\boldsymbol{X}_{k+1}^i$ and $\boldsymbol{X}_k^i$.

38

SDM: training

casa Paganini infomus

- We now want to learn R_{k+1} and $\boldsymbol{b}_{k+1}$, which will lead us to $\boldsymbol{x}_{k+2}^i$. For each face $\boldsymbol{d}^i$, we have that:
  - $\boldsymbol{X}_{k+2}^i = \boldsymbol{X}_{k+1}^i + \mathrm{R}_{k+1}\boldsymbol{\Phi}_{k+1}^i + \boldsymbol{b}_{k+1}$, being $\boldsymbol{\Phi}_{k+1}^i = \boldsymbol{h}(\boldsymbol{X}_{k+1}^i)$
  - The error we make when we move to $\boldsymbol{x}_{k+2}^i$ thus is:

$$e(\boldsymbol{X}_{k+1}^i) = \left\|\boldsymbol{x}_*^i - \boldsymbol{X}_{k+2}^i\right\|^2 = \left\|\boldsymbol{x}_*^i - (\boldsymbol{X}_{k+1}^i + \mathrm{R}_{k+1}\boldsymbol{\Phi}_{k+1}^i + \boldsymbol{b}_{k+1})\right\|^2$$
$$= \left\|\boldsymbol{x}_*^i - \boldsymbol{X}_{k+1}^i - \mathrm{R}_{k+1}\boldsymbol{\Phi}_{k+1}^i - \boldsymbol{b}_{k+1}\right\|^2 =$$
$$= \left\|\Delta\boldsymbol{X}_{*,k+1}^i - \mathrm{R}_{k+1}\boldsymbol{\Phi}_{k+1}^i - \boldsymbol{b}_{k+1}\right\|^2$$

being $\Delta\boldsymbol{X}_{*,k+1}^i = \boldsymbol{x}_*^i - \boldsymbol{X}_{k+1}^i$.

39

SDM: training and testing

casa Paganini infomus

- We can apply the same method we used for R_0 and $\boldsymbol{b}_0$, so that R_{k+1} and $\boldsymbol{b}_{k+1}$ can be learnt by minimizing:

$$\arg\min_{\mathrm{R}_{k+1},\,\mathbf{b}_{k+1}} \sum_{i=1}^{N} \sum_{\boldsymbol{x}_{k+1}^i} \left\|\Delta\boldsymbol{x}_{*,k+1}^i - \mathrm{R}_{k+1}\boldsymbol{\phi}_{k+1}^i - \boldsymbol{b}_{k+1}\right\|^2$$

- The algorithm often converges in 4 or 5 steps.
- During the testing phase, the algorithm starts from an initial position $\boldsymbol{x}_0$ and applies the sequence of displacements $\{\Delta\boldsymbol{x}_k\}_{k=1\ldots\mathrm{K}}$ (being K the number of steps needed to converge) to find the actual landmark positions.

40

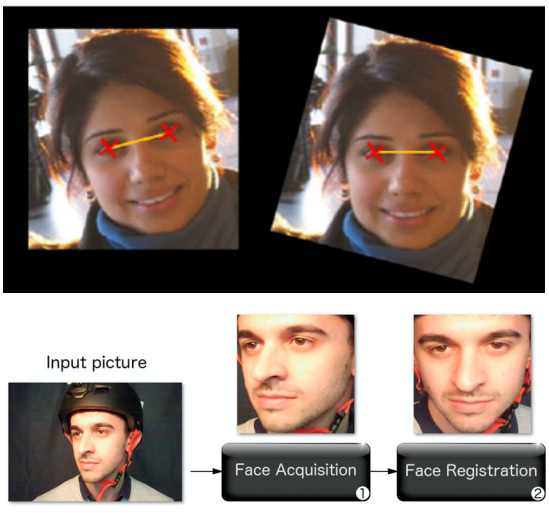
SDM: examples

Example results on LFW dataset (http://vis-www.cs.umass.edu/lfw/)
(Figure from Xiong and De la Torre, 2013)

41

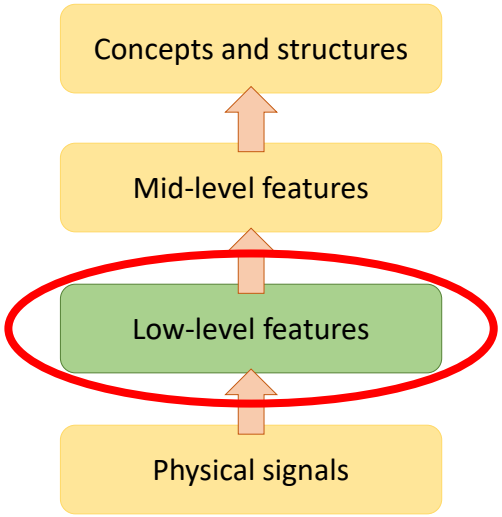
Face registration

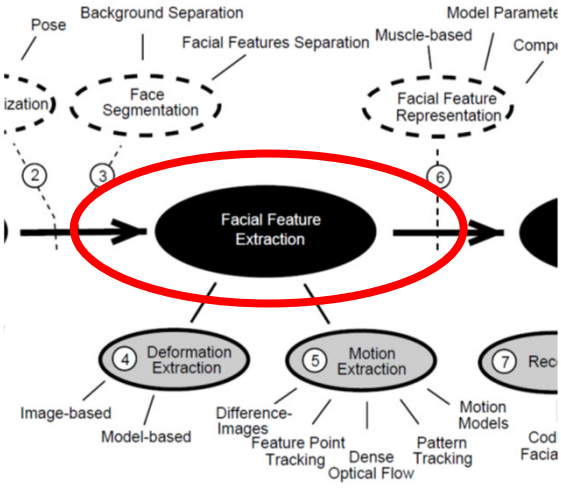
- Misalignments produce large variations in the face appearance, resulting in large intra-class variance.
- Face registration refers to adjusting faces with respect to a common pre-defined reference coordinate system.
- Methods include Procustes transformation and piecewise affine transformation.



42

A conceptual framework





43

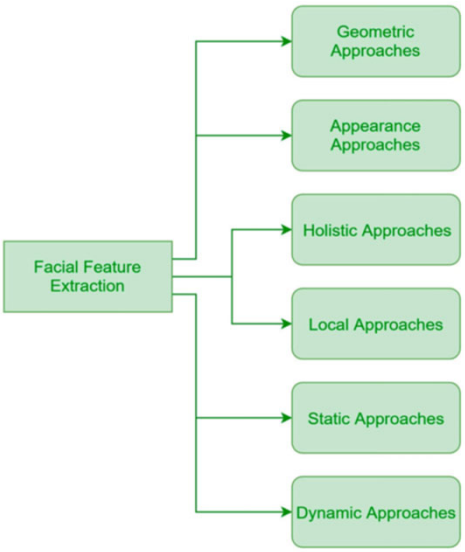
Facial features

- Fasel and Luettin (2003) distinguish between features focusing on:
  - Deformation.
  - Motion.
- More recently, Martinez and colleagues (2019) grouped feature extraction methods into four categories:
  - Appearance-based.
  - Geometry-based.
  - Motion-based.
  - Hybrid methods.
- Moreover, features can be local or holistic.

44

Facial features

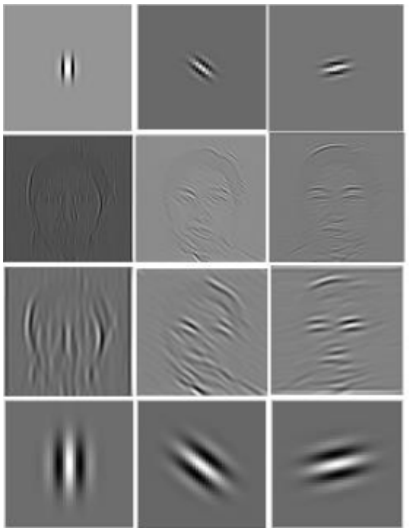
- Fei and colleagues (2019) categorize approaches to computation of facial features along three axes:
  - Geometric vs. appearance approaches.
  - Holistic vs. local approaches.
  - Static vs. dynamic approaches.



45

Appearance features

- Appearance features describe the color and texture of a facial region and can be divided into five major categories (Martinez et al., 2019):
  - Image intensity descriptors.
  - Descriptors computed by using filter banks.
  - Binarized local texture descriptors.
  - Gradient-based descriptors.
  - Two-layers descriptors.



Picture from (Goyani and Patel, 2017)

46

Appearance features

casa Paganini infomus

- Appearance features are computed (Martinez et al., 2019):
  - On the whole face (holistic features).
  - By adopting a block-based approach (tiling).
  - On selected regions around relevant points of the face (Region Around Point, RAPs).
  - On selected regions defined by points (Region Of Interest, ROIs).

Picture from (Martinez et al., 2014).

47

Image intensity descriptors

casa Paganini infomus

- Image intensity descriptors consist of the histogram or other statistical descriptors of the raw pixel intensities, computed on the selected features after face registration.
- Advantages :
  - Easy to implement
  - Quick to compute.
- Limitations :
  - Very sensitive to all kinds of distractor variation.
  - Non-frontal head-poses are problematic.
  - They often need to be combined with other descriptors.

48

Filter banks

casa Paganini infomus

- Features are obtained by convolving a region of the input face with a set of filters (a filter bank).
- Filter banks are used for analyzing deformations, i.e., shape and texture changes leading to high spatial gradients.
- Gabor filters are commonly used in face analysis.

49

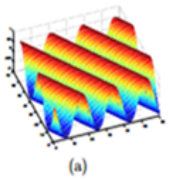
Filter banks: Gabor filters

casa Paganini infomus

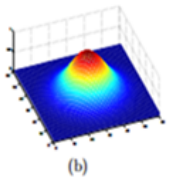
- Gabor filters are special classes of band pass filters that can be viewed as a sinusoidal signal of a particular frequency and orientation, modulated by a Gaussian wave.

$$g_e(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}(\frac{x^2}{\sigma_x}+\frac{y^2}{\sigma_y})} \cos(2\pi\omega_{x_0}x + 2\pi\omega_{y_0}y)$$

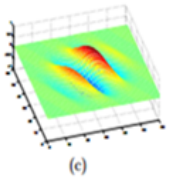
$$g_o(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}(\frac{x^2}{\sigma_x}+\frac{y^2}{\sigma_y})} \sin(2\pi\omega_{x_0}x + 2\pi\omega_{y_0}y)$$



A Sinusoid oriented 30° with X-axis



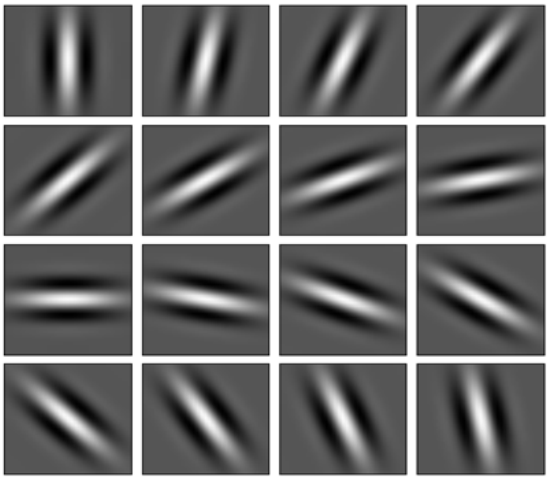
A 2-D Gaussian



The corresponding 2-D Gabor filter

50

Filter banks: Gabor filters

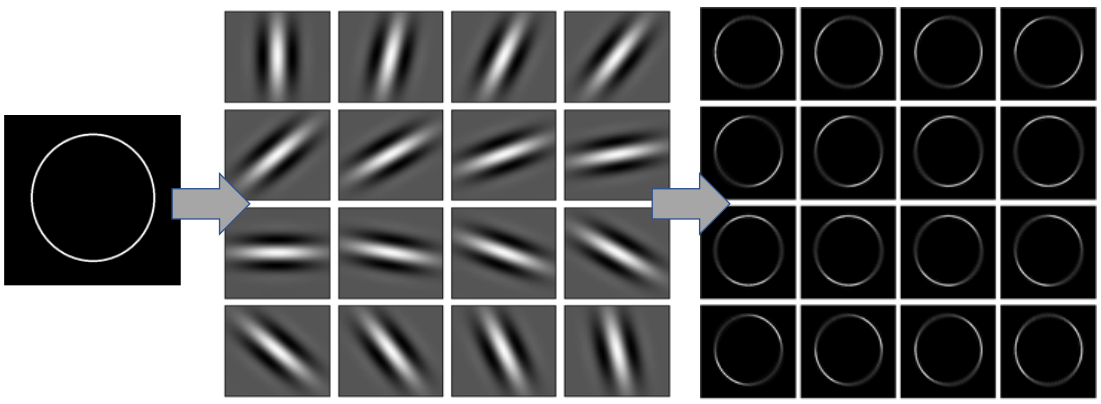


A bank of 16 Gabor filters oriented at angles of 11.250° apart (i.e., if the first filter is at 0°, then the second will be at 11.250°, the third will be at 22.50°, and so on).

- The filter bank is made of Gabor filters with different orientations.
- Consider, e.g., a white circle in black background. When this image is passed through each filter in the filter bank, the edge of the circle which gets detected is the edge oriented at an angle at which one of Gabor filters is oriented.

51

Filter banks: Gabor filters

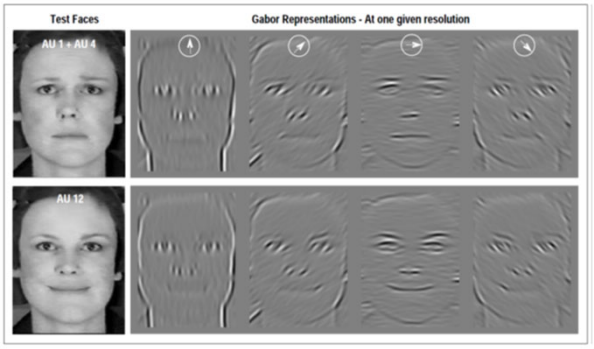


Input image | Bank of 16 Gabor filters | Output of each filter

52

Filter banks: Gabor filters

- Input images (faces or regions) are given to a bank of Gabor filters, i.e., they are convolved with the filters in the bank.
- The magnitude values of the complex response of each filter, i.e., for each orientation and central frequency, are concatenated into a feature vector.



Gabor representations of two facial expression obtained with a bank of 4 Gabor filters. Picture from (Fasel and Luettin, 2003).

53

Filter banks: Gabor filters

- Advantages :
  - They can be a powerful representation, provided that the parametrization is correct.
  - They are robust to small registration errors.
- Limitations :
  - The resulting dimensionality is very large.
  - Computational cost is high.
  - Consequently, it is hard to make them work in real-time.

54

Binarized local texture descriptors

casa Paganini infomus

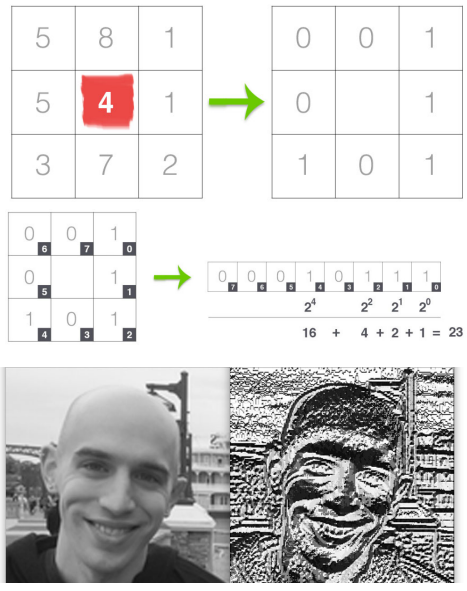
- Two major approaches:
  - Local Binary Patterns (LBP).
  - Local Phase Quantization (LPQ).
- Many works on facial expression successfully used LBP features in a block-based holistic manner and 10 × 10 blocks were found to be a suitable choice.
- LPQs were also frequently applied in a block-based holistic manner, 4 × 4 blocks appeared to be a suitable choice.
- LBP and LPQ descriptors may also be combined.

55

Local Binary Patterns

casa Paganini infomus

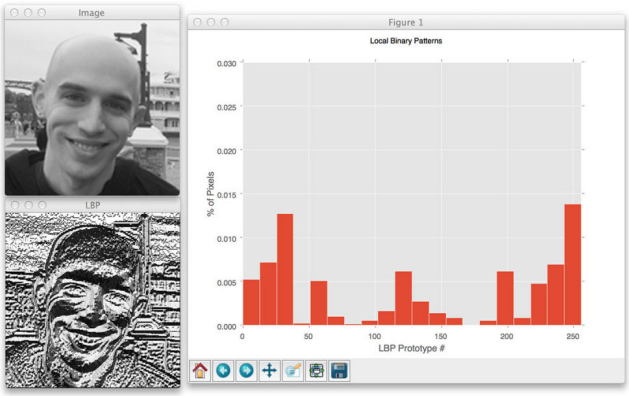
- LBPs are texture descriptors (e.g., see Ojala et al., 2002).
- An 8-dimensional binary vector is taken for each pixel.
- Each binary value encodes whether the intensity of the central pixel is larger than each of the neighboring pixels.
- The intensity value of each pixel in the output LBP image is the decimal representation of the pixel's LBP.



56

Local Binary Patterns

- A histogram is then computed, where each bin corresponds to one of the different possible binary patterns, resulting in a 256-dimensional descriptor.
- The histogram is treated as a feature vector.

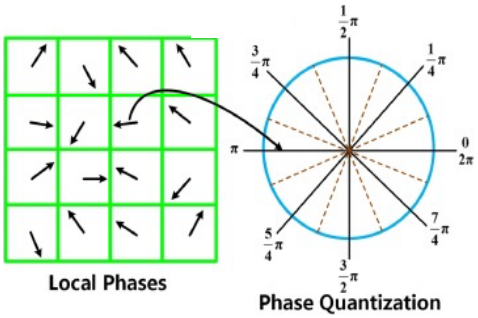


Pictures from:
https://www.pyimagesearch.com/2015/12/07/local-binary-patterns-with-python-opencv/

57

Local Phase Quantization

- It uses local phase information extracted by using a 2D Fourier transform computed over a M × M neighborhood at each pixel position (e.g., see Ojansivu and Heikkilä 2008).
- The phase information is quantized by keeping the signs of the real and imaginary parts of each component.

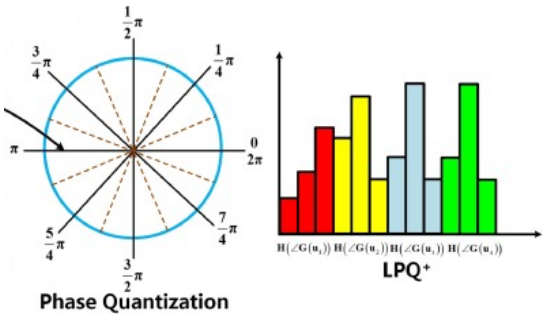


58

Local Phase Quantization

casa Paganini infomus

- The quantized coefficients are represented as integer values between 0-255 using binary coding.
- Finally, a histogram of these integer values from all image positions is composed and used as a 256-dimensional feature vector.

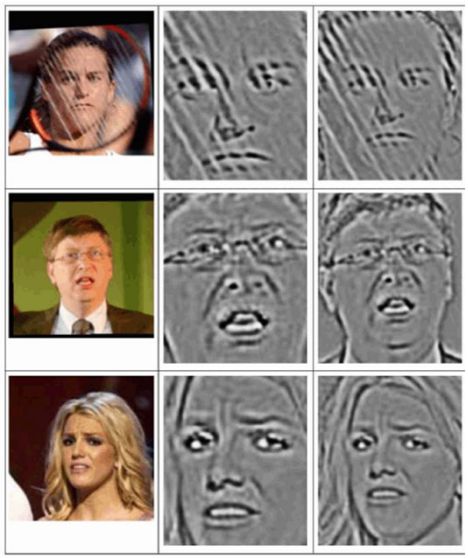


59

Binarized local texture descriptors

casa Paganini infomus

- Advantages:
  - Robust to illumination conditions.
  - Robust to misalignments.
  - Suitable for holistic representations.
  - Computational simplicity.
- Limitations :
  - LBPs are not robust to rotations.
  - LBPs need a correct normalization of the face to an upright position in order to work properly.



60

Gradient-based descriptors

- They use a histogram to encode the gradient information of the represented image (or a portion of image).
- Commonly applied approaches include Histogram of Oriented Gradients (HOG), Scale-Invariant Feature Transform (SIFT), and DAISY.

Example of Histogram of Oriented Gradients (HOG)

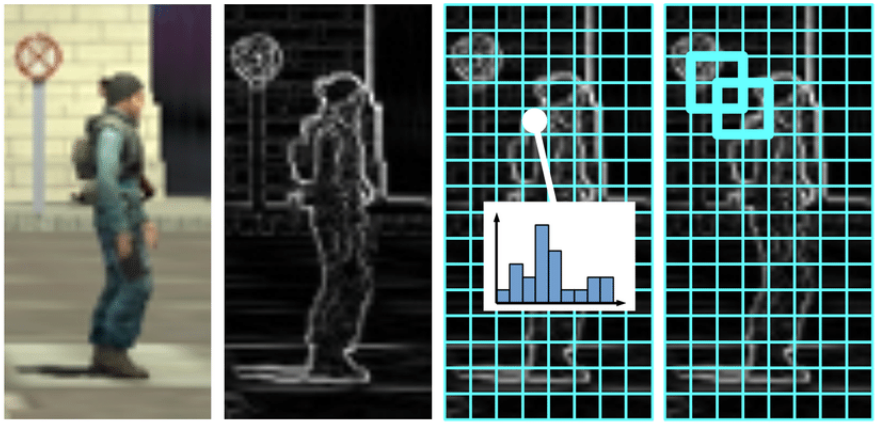
61

Histogram of oriented gradients

- HOGs were introduced by R. K. McConnell in 1986. Steps:
  - Horizontal and vertical gradients are computed by filtering the image with the following kernels: [-1, 0, 1] and $[-1, 0, 1]^T$.
  - The magnitude and phase of gradient is computed for each pixel.
  - The image is divided into blocks (64 × 128 blocks) and a histogram of gradients is calculated for each block.
  - Blocks are grouped in bigger blocks (e.g., 4 blocks are grouped by using a sliding window approach with overlap): the histograms from the smaller blocks are concatenated and normalized.
  - The normalized histograms from the bigger blocks are concatenated into one single vector treated as a feature vector.

62

Histogram of oriented gradients



detection window slides over an image

at each location where the window is applied, gradients are computed

window is evently partitioned into cells and each pixel of the cell contirbutes to cell gradient orientation histogram

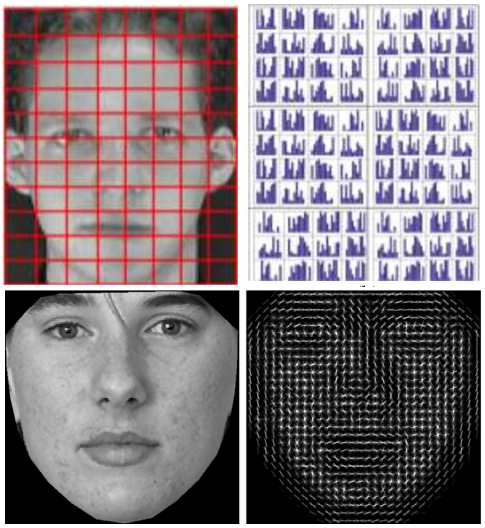
orientation histograms for overlapping 2x2 blocks of cells are normalized and collected to form the final descriptor

Picture from (Olejniczak and Kraft, 2017).

63

Gradient-based descriptors

- Advantages :
  - Robust to misalignment.
  - Robust to uniform illumination variations.
  - Robust to affine transformations.
- Limitations:
  - They need to be applied locally to avoid larger gradients dominating the representation.



64

Two-layers descriptors

casa Paganini infomus

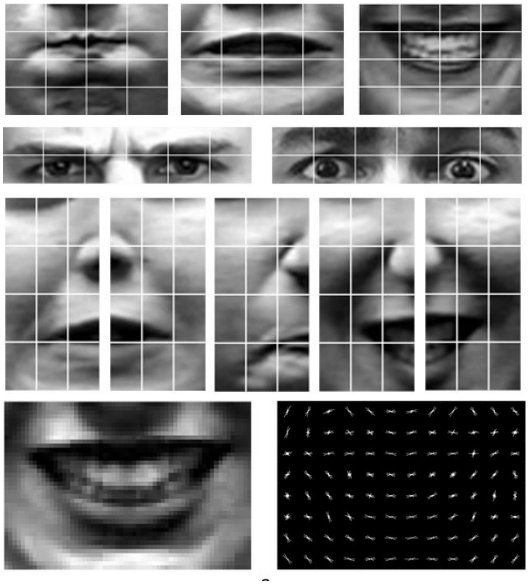
- Two-layer appearance descriptors result from the application of two traditional feature descriptors, where the second descriptor is applied over the response of the first one (Martinez et al., 2019).
- An example, Local Gabor Binary Patterns (LGBP, e.g., see Senechal et al., 2012):
  - LGBPs are computed by first calculating Gabor magnitudes over the image and then applying an LBP operator over the resulting Gabor response maps.
  - Gabor features are applied first to capture local structures.
  - The LBP operator increases the robustness to misalignment and illumination changes and reduces feature dimensionality.

65

Appearance features

casa Paganini infomus

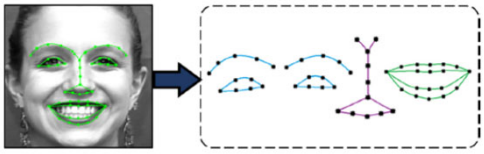
- Advantages:
  - They are flexible and can be extracted from the whole face (holistic features) or from face regions (local features).
  - They are nowadays the most commonly used features.
- Limitations:
  - They can be sensitive to non-frontal head poses.
  - They can be sensitive to illumination changes.



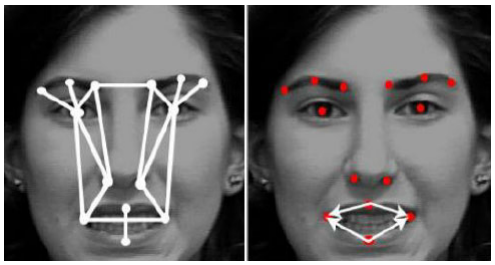
66

Geometric features

- Geometric features capture statistics derived from location of facial landmarks, with most facial muscle activations resulting in their displacement (Martinez et al., 2019).
- I.e., they measures variations e.g., in shape, location, and distance of relevant facial regions such as mouth, eyes, eyebrows, nose, and so on.



Location of landmarks

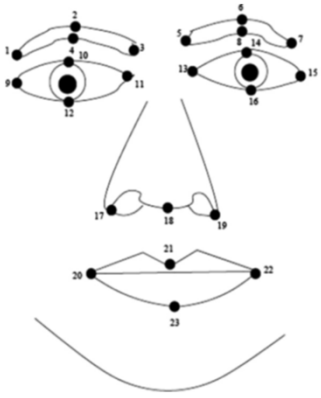


Distances Angles

67

Geometric features

- Example: Majumder and colleagues (2013) use an analytical model consisting of 23 facial points.



Geometric features	Number of features	Description
Eyebrow	4x2	Two extreme corners, upper and lower mid points
Eyes	4x2	Two extreme corners, upper and lower mid points
Nose	3	Two nostrils and nose tip
Lip	4	Two extreme corners, upper and lower mid points

Picture from (Majumder et al., 2014).

Majumder, A., Behera, L., Subramanian, V. K., 2014. Emotion recognition from geometric facial features using self-organizing map. Pattern Recognition, 47, 3.

68

Geometric features

casa Paganini infomus

- Starting from the model, they compute 26 dimensional geometric features consisting of displacement of 8 eyebrow points and 4 lip points along x- and y-direction, and projection ratios of two eyes.
- Displacement is calculated by using the neutral expression as reference.

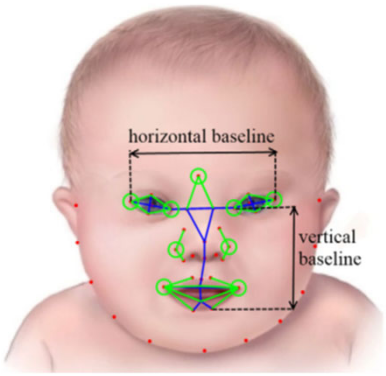
Picture from (Majumder et al., 2014).

69

Geometric features

casa Paganini infomus

- Another example: Zhao and colleagues (2013) use 27 geometric features consisting of Euclidean distances between landmarks and corner angles spanned by landmarks:
  - Euclidean distances are divided to horizontal and vertical lines according to their directions.
  - The horizontal and vertical lines are both normalized by their baselines.
  - Angles are analyzed via linear statistics.

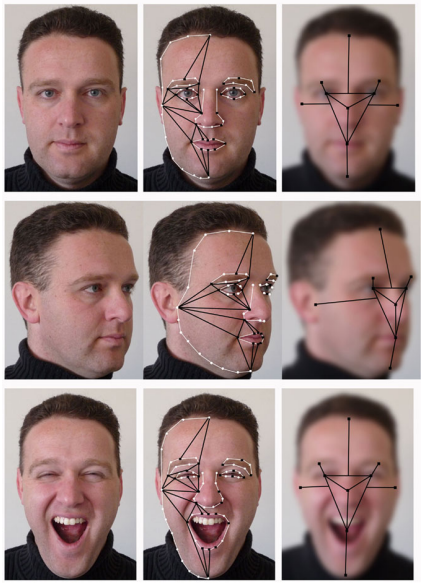


Picture from (Zhao et al., 2013).

Zhao, Q., Okada, K., Rosenbaum, K., Kehoe, L., Zand, D., Sze, R., Summar, M., Linguraru, M. G., 2014. Digital Facial Dysmorphology for Genetic Screening: Hierarchical Constrained Local Model Using ICA. Medical Image Analysis. 18.

70

Geometric features

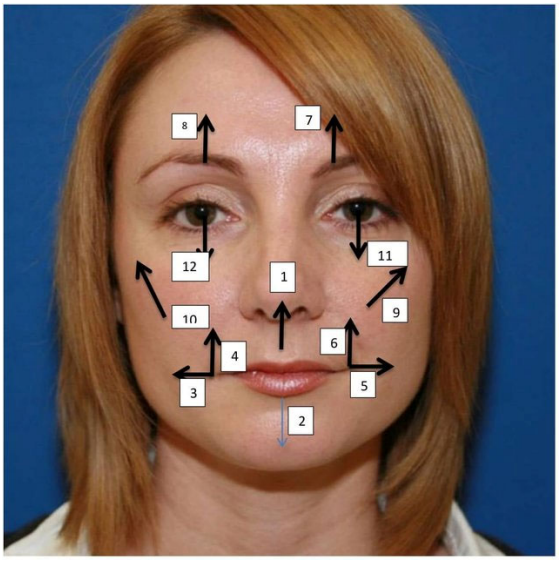


- Advantages:
  - Easy to register.
  - Independent of lighting conditions.
- Limitations:
  - They are unable to capture facial expressions that do not cause landmark displacements.
- To overcome limitations geometric features are often combined with appearance features.

71

Motion features

- Motion features capture flexible deformations of the skin caused by the contraction of facial muscles (Martinez et al., 2019).
- Motion extraction methods include:
  - Difference-images.
  - Optical flow.

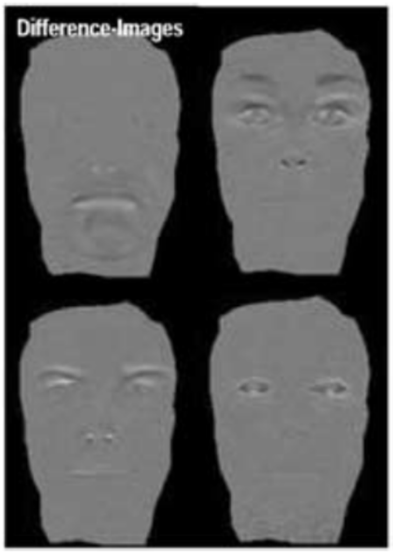


72

Difference images

casa Paganini infomus

- Difference images (or δ-images) are defined as the pixel-wise difference between the current frame and an expressionless-face frame of the same user.
- The first frame of the facial expression is often taken as the neutral, expressionless-face to compute the difference with.
- Difference images are then further analyzed for computing features.



Picture from (Fasel and Luettin, 2003).

73

Difference images

casa Paganini infomus

- Originally conceived for analysis of full-body movement, Motion History Images (MHIs) and Motion Energy Images (MEIs) use image differences to summarize motion over time.
- MEIs are binary images that indicate whether any pixel differences occurred over a number of frames.
- MHIs represent recent motion by high intensity values, while the pixels where motion was detected longer ago fade to zero intensity linearly over time.

74

Optical flow

casa Paganini infomus

- Concept introduced by psychologist James J. Gibson in the 1940s to describe the visual stimulus provided to animals moving through the world.
- It is defined as the pattern of apparent motion of objects, surfaces, and edges in a visual scene caused by the relative motion between an observer and a scene (Gibson, 1950).
- It can also be defined as the distribution of apparent velocities of movement of brightness patterns in an image (Horn and Rhunck, 1981).
- Concretely, it can be though as a measure of the direction and intensity of the motion for every pixel in an image.

75

Optical flow

casa Paganini infomus

- Let us take the pixel at location (x_0, y_0) having intensity $I(x_0, y_0, t_0)$ at time t_0.
- Let us assume that at time $t_0 + \Delta t$ the pixel moves by Δx and Δy along the two spatial dimensions, respectively.
- The following *brightness constancy constraint* can be given:

$$I(x_0, y_0, t_0) = I(x_0 + \Delta x, y_0 + \Delta y, t_0 + \Delta t)$$

- Assuming movement is small, $I(x_0 + \Delta x, y_0 + \Delta y, t_0 + \Delta t)$ can be developed with Taylor series to get:

$$I(x_0 + \Delta x, y_0 + \Delta y, t_0 + \Delta t) \approx I(x_0, y_0, t_0) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t$$

76

Optical flow

casa Paganini infomus

- By applying the brightness constancy constraint, it follows that:

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0$$

- By multiplying both sides by the factor $1/\Delta t$, we get:

$$\frac{\partial I}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y}\frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t}\frac{\Delta t}{\Delta t} = 0$$

- We note now that $\Delta x/\Delta t$ and $\Delta y/\Delta t$ are the horizontal and vertical components of velocity v_x and v_y respectively, so that:

$$\frac{\partial I}{\partial x}v_x + \frac{\partial I}{\partial y}v_y + \frac{\partial I}{\partial t} = 0$$

77

Optical flow

casa Paganini infomus

- Let us call the three partial derivatives as follows:

$$I_x = \frac{\partial I}{\partial x}(x_0, y_0, t_0),\ \ I_y = \frac{\partial I}{\partial y}(x_0, y_0, t_0),\ \ I_t = \frac{\partial I}{\partial t}(x_0, y_0, t_0)$$

- Then:

$$I_x v_x + I_y v_y = -I_t$$

- That is :

$$\nabla I \cdot \boldsymbol{v} = -I_t$$

- This is an equation (the optical flow equation) in two unknowns (v_x and v_y) and cannot be solved as such.

78

Optical flow

casa Paganini infomus

- This is known as the aperture problem of optical flow: to compute v_x and v_y another set of equations is needed, given by some additional constraint.
- Thus, optical flow methods introduce additional conditions for estimating the actual flow and can be distinguished depending on the kind of additional constraints they adopt:
  - Phase correlation methods.
  - Block-based methods.
  - Differential methods.
  - Discrete optimization methods.

79

Lucas–Kanade method

casa Paganini infomus

- One of the most commonly used methods for optical flow, first proposed by Bruce D. Lucas and Takeo Kanade in 1981.
- Additional constraint: the pixels in the neighborhood of (x_0, y_0) have the same velocity components v_x and v_y.
- Let us take n pixels $(x_1, y_1)\ldots(x_n, y_n)$ in the neighborhood of (x_0, y_0), so that:

$$I_{jx} = \frac{\partial I}{\partial x}(x_j, y_j, t_0),\ \ I_{jy} = \frac{\partial I}{\partial y}(x_j, y_j, t_0),\ \ I_{jt} = \frac{\partial I}{\partial t}(x_j, y_j, t_0)$$

for each $j = 1 \ldots n$.

80

Lucas–Kanade method

casa Paganini infomus

- Then, we can define:

$$\mathrm{A} = \begin{bmatrix} I_{1x} & I_{1y} \\ \vdots & \vdots \\ I_{nx} & I_{ny} \end{bmatrix}, \qquad \boldsymbol{v} = \begin{bmatrix} v_x \\ v_y \end{bmatrix}, \qquad \boldsymbol{b} = \begin{bmatrix} -I_{1t} \\ \vdots \\ -I_{nt} \end{bmatrix}$$

- This system has more equations than unknowns and thus it is usually over-determined. The Lucas–Kanade method obtains a compromise solution by applying the least squares principle, that is by computing:

$$\min_{\boldsymbol{v}} \|A\boldsymbol{v} - \boldsymbol{b}\|^2$$

81

Lucas–Kanade method

casa Paganini infomus

- It can be proved that the minimum least squares solution is given by the solution (in $\boldsymbol{v}$) of:

$$A^T A\boldsymbol{v} = A^T\boldsymbol{b}$$

- That is:

$$\boldsymbol{v} = \left(A^T A\right)^{-1} A^T\boldsymbol{b}$$

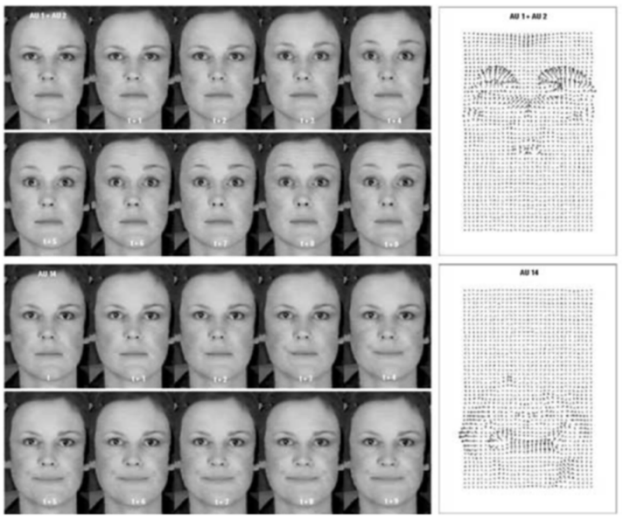
- Note that $\left(A^T A\right)^{-1}$ should be invertible, or its eigenvalues $\lambda_1$and λ_1 should satisfy condition $\lambda_1 \geq \lambda_2 > 0$.
- Further, to avoid noise, usually λ_2 is required not to be too small.

82

Optical flow

casa Paganini infomus

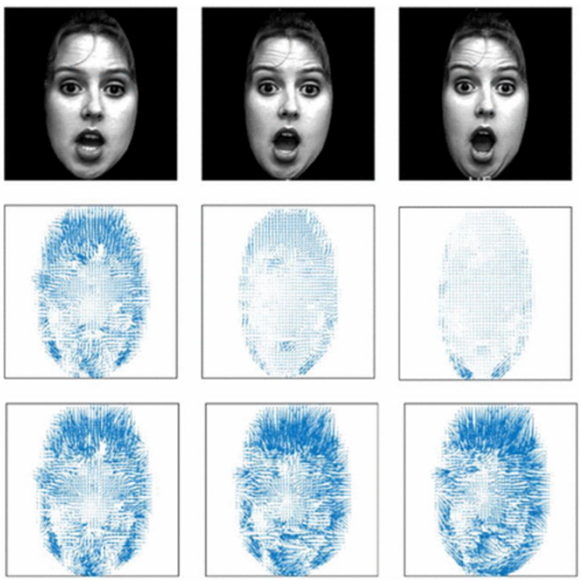
- The average normalized velocity of each pixel, computed over time during a facial expression or a part of it, can be taken as feature vector.
- This will produce two numbers ($\overline{v_x}$ and $\overline{v_y}$) for each pixel, that is in very large feature vectors.



83

Motion features

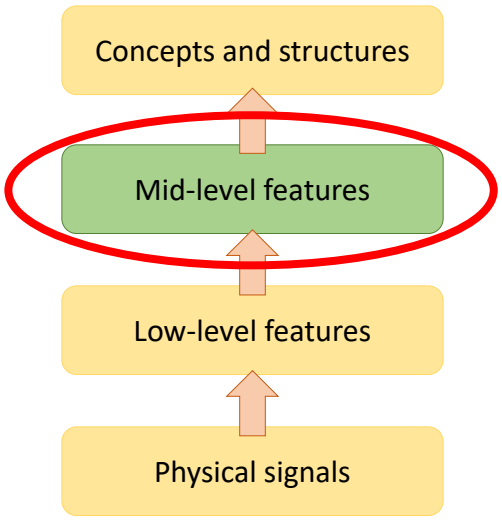
casa Paganini infomus

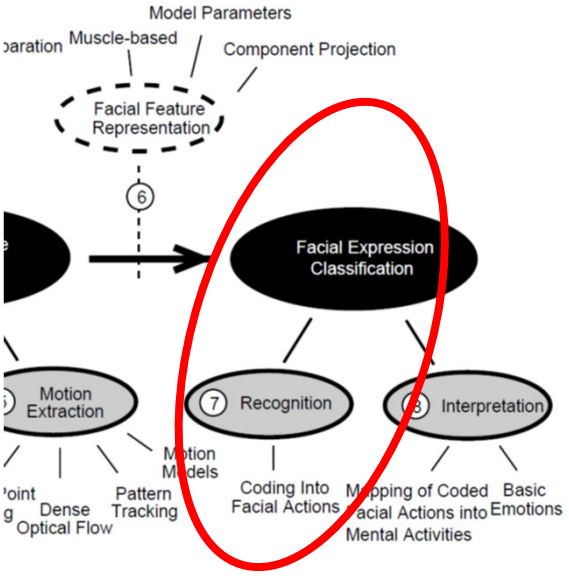


- Advantages:
  - Less person specific than appearance features.
- Limitations:
  - They require full elimination of rigid motion.
  - They are affected by misalignment and varying illumination conditions.

84

A conceptual framework





85

Face action units

- Facial expression can be described by means of discrete face action units and their dynamics.

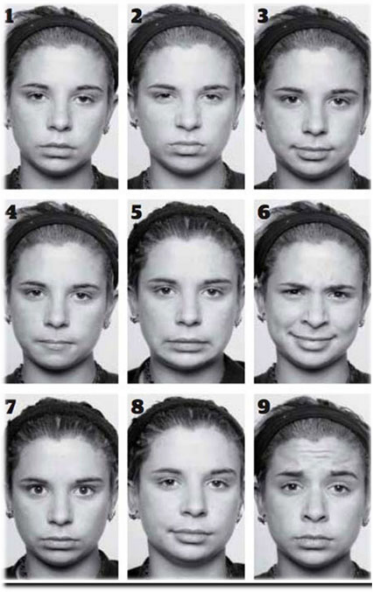
Examples of face action units

86

Face action units

casa Paganini infomus

- The Facial Action Coding System (FACS) is a system to taxonomize human facial movements by their appearance on the face.
- It encodes movements of individual facial muscles.
- FACS is based on a system originally developed by Swedish anatomist Carl-Herman Hjortsjö.
- Later adopted by P. Ekman and W. V. Friesen, published in 1978, revised in 2002.



87

Face action units

casa Paganini infomus

- FACS include:
  - Action Units (AUs): 32 atomic facial muscle actions performed by either individual muscles or groups of muscles.
  - Action Descriptors (ADs): 14 additional items accounting for head pose, gaze direction, and miscellaneous actions (e.g., jaw thrust, blow and bite).
  - Intensities of AUs: annotated by appending a letter (A to E) to the Action Unit number:

    A: Trace
    B: Slight
    C: Marked or Pronounced
    D: Severe or Extreme
    E: Maximum

88

Face action units

casa Paganini infomus

- FACS describes morphology and dynamics of a facial display.
- Morphology refers to facial configuration and is observed from static frames. This is encoded in the detected AUs.
- Dynamics reflect the temporal evolution of one facial display to another and is observed from videos.
- Facial dynamics is encoded in the timing, duration, and speed of activation and deactivation of various AUs.
- Dynamics can be explicitly analyzed by detecting the boundaries of the temporal phase (namely neutral, onset, apex, offset) of each AU activation.

89

Face action units

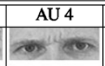
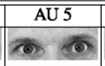
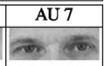
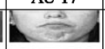
casa Paganini infomus

- Using FACS, nearly any anatomically facial expression is coded by deconstructing it into specific Action Units and their temporal segments producing the expression.
- Action Units are independent of any interpretation and can be used for any higher order decision making process, e.g., recognition of emotions.
- FACS is usually applied by manual annotators. It takes over 100 hours of training to achieve minimal competency as a FACS coder, and each minute of video takes approximately one hour to score (Martinez et al., 2019).

90

Face action units

casa Paganini infomus

Upper Face Action Units					
AU 1	AU 2	AU 4	AU 5	AU 6	AU 7
					
Inner Brow Raiser	Outer Brow Raiser	Brow Lowerer	Upper Lid Raiser	Cheek Raiser	Lid Tightener
*AU 41	*AU 42	*AU 43	AU 44	AU 45	AU 46
					
Lid Droop	Slit	Eyes Closed	Squint	Blink	Wink
Lower Face Action Units					
AU 9	AU 10	AU 11	AU 12	AU 13	AU 14
					
Nose Wrinkler	Upper Lip Raiser	Nasolabial Deepener	Lip Corner Puller	Cheek Puffer	Dimpler
AU 15	AU 16	AU 17	AU 18	AU 20	AU 22
					
Lip Corner Depressor	Lower Lip Depressor	Chin Raiser	Lip Puckerer	Lip Stretcher	Lip Funneler
AU 23	AU 24	*AU 25	*AU 26	*AU 27	AU 28
					
Lip Tightener	Lip Pressor	Lips Part	Jaw Drop	Mouth Stretch	Lip Suck

91

Face action units

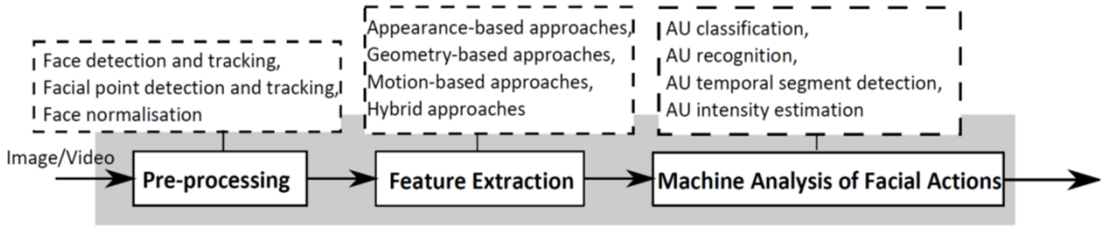
casa Paganini infomus

Examples of upper and lower face AUs defined in the FACS. Picture from (Martinez et al., 2019).

92

Automatic coding of AUs

- Manually coding of AUs is time-consuming.
- Consequently, automatic systems were developed for coding AUs from videos. These usually include face detection, computation of facial features, and machine learning.



Configuration of a generic facial action recognition system. Picture from (Martinez et al., 2019).

93

Automatic coding of AUs

- Four major problems can be distinguished:
  - AU detection: it produces a binary frame-level label per target AU, indicating whether the AU is active or not.
  - AU intensity estimation: it infers frame-level labels of intensity per target AU, as described in the FACS manual (i.e., A, B, C, D, or E).
  - AU temporal segment detection: it infers frame-level labels of temporal segment per target AU, as described in the FACS manual (i.e., neutral, onset, apex, offset).
  - AU classification: this is uncommon nowadays, and deals with sequences containing pre-segmented AU activation episodes.

94

Automatic coding of AUs

casa Paganini infomus

- The output for each class of problems can be summarized as:

Problem	Variants	Output space
Class.	No AU Co-ocur.	$\mathcal{Y} = \{1 : k\}$ per seq.
	AU Co-ocurence	$\mathcal{Y} = \{\pm 1\}^k$ per seq.
Detection	Frame-based inf.	$\mathcal{Y} = \{\pm 1\}^k$ per fr.
	Segment-based inf.	
Intensity	Multiclass	$\mathcal{Y} = \{0 : 5\}^k$ per fr.
	Ordinal reg.	
	Regression	$\mathcal{Y} = [0, 5]^k$ per fr.
Temp. seg.	Class.	$\mathcal{Y} = \{0 : 3\}^k$ per fr.

k indicates the number of AUs considered.

Table from (Martinez et al., 2019).

95

An example of AU detection

casa Paganini infomus

- The Automated Facial Action Coding System by Hamm, Kohler, Gur, and Verma (J. Neurosci. Methods, 2011).
- Face detection: Viola and Jones face detector is applied, Active Shape Model (ASM) is used for localizing 159 landmarks.

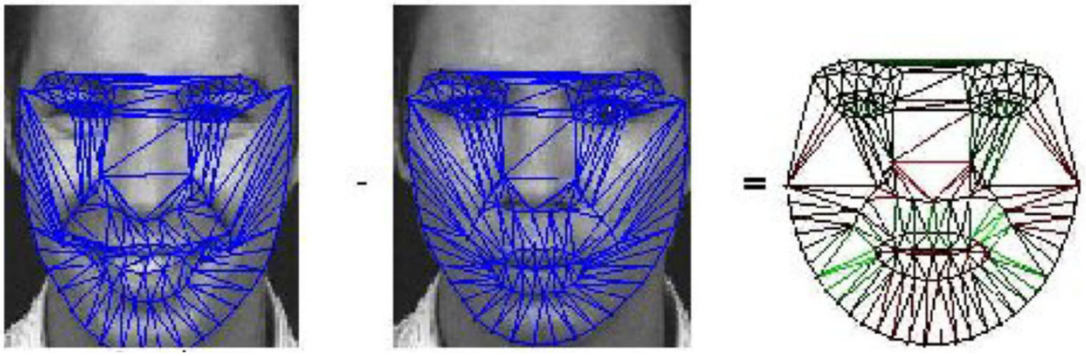
Results of face detection and landmark localization (picture from Hamm et al., 2011)

96

An example of AU detection

casa Paganini infomus

- Geometric features: the deformation of the face mesh relative to the mesh at a neutral state is computed. Compression and expansion of the edges is measured.



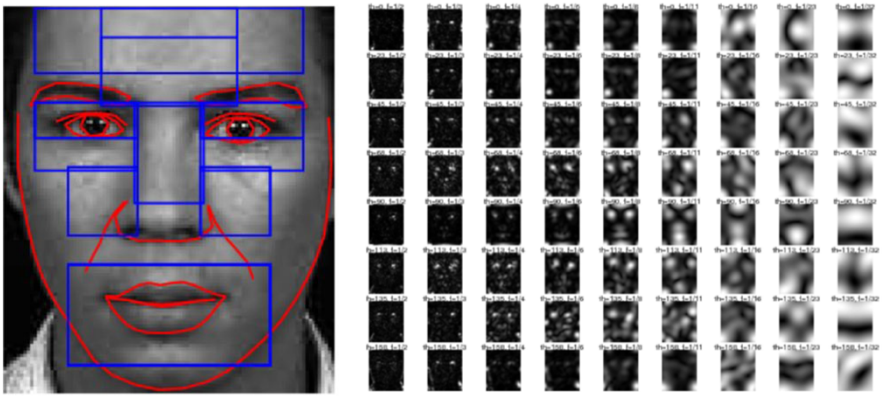
Green and red edges on the template mesh indicate compression and expansion of the edges, respectively. Picture from (Hamm et al., 2011)

97

An example of AU detection

casa Paganini infomus

- Appearance features: Gabor filters are applied to relevant facial regions. Difference with the neutral face is computed.



Gabor filters are applied to the blue rectangle regions. Picture from (Hamm et al., 2011).

98

An example of AU detection

casa Paganini infomus

- Training set: 3419 faces.
- 15 AdaBoost classifiers trained to detect each of the 15 AUs independently (4 AUs discarded for lacks of samples).
- Accuracy ranging from 87% to over 99%.
- Overall accuracy of 95.9%.

AU No	Description	Rate (%)
AU1	Inner Brow Raiser	95.8
AU2	Outer Brow Raiser	97.8
AU4	Brow Lowerer	91.0
AU5	Upper Lid Raiser	96.9
AU6	Cheek Raiser	93.0
AU7	Lid Tightener	87.0
AU9	Nose Wrinkler	97.5
AU10	Upper Lip Raiser	99.3
AU12	Lip Corner Puller	97.1
AU15	Lip Corner Depressor	99.2
AU17	Chin Raiser	96.5
AU18	Lip Puckerer	98.6
AU20	Lip Stretcher	97.7
AU23	Lip Tightener	96.9
AU25	Lips Part	95.7

99

Automatic coding of AUs

casa Paganini infomus

- Many other approaches were explored, e.g.:

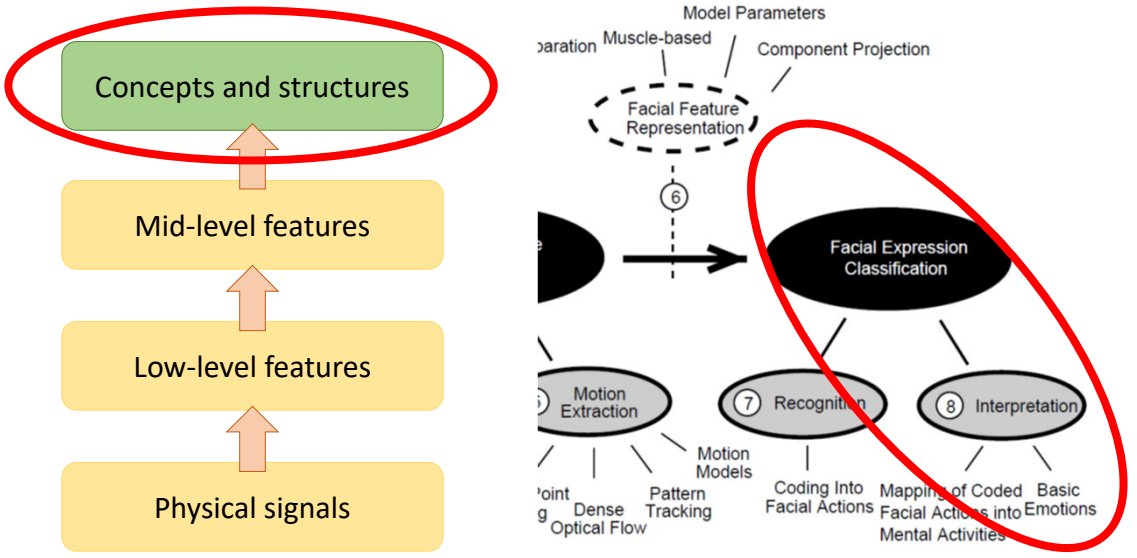
Table 1 Overview of the two main components of systems for action unit recognition

Study	Features	Classifier
Lien et al. (1998)	Dense-flow tracking	Hidden Markov model
Cohn et al. (1999)	Tracked feature points	Quadratic discriminant classifier
Fasel and Luettin (2000)	Eigenfaces	Nearest neighbor classifier
Bartlett et al. (2005, 2006)	Gabor filters	Boosting + support vector machine
Chang et al. (2006)	Manifold learning	Bayesian
Whitehill and Omlin (2006)	Haar features	Boosting
Littlewort et al. (2006)	Gabor filters	Boosting + support vector machine
Lucey et al. (2007)	Active appearance model	Support vector machine
Valstar et al. (2004)	Motion history images	Nearest neighbor classifier
Pantic and Rothkrantz (2004)	Tracked feature points	Rule base
Pantic and Patras (2005)	Tracked feature points	Rule base
Valstar and Pantic (2006, 2007)	Tracked feature points	Boosting + support vector machine
Tong et al. (2007, 2010)	Gabor filters	Boosting + dynamic bayesian network
Susskind et al. (2008)	Normalized pixels	Deep belief network
Koelstra et al. (2010)	Free-form deformations	Boosting + hidden Markov model

Table from (van der Maaten and Hendriks, Cogn Process, 2011)

100

A conceptual framework



101

Facial expression recognition

- Interpretation of facial expression via algorithms.
- Depending on context, facial expressions may have varied communicative functions. They can regulate conversations by signaling turn-taking, convey biometric information, express intensity of mental effort, and signal emotion.
- Most existing facial expression recognizers focus on emotion.

Recognizer → Anger

102

Facial expression recognition

casa Paganini infomus

- Action Units are directly associated with facial expressions.

	AUs
FACS:	upper face: 1, 2, 4-7, 43, 45, 46; lower face: 9-18, 20, 22-28; other: 21, 31, 38, 39
anger:	4, 5, 7, 10, 17, 22-26
disgust:	9, 10, 16, 17, 25, 26
fear:	1, 2, 4, 5, 20, 25, 26, 27
happiness:	6, 12, 25
sadness:	1, 4, 6, 11, 15, 17
surprise:	1, 2, 5, 26, 27
pain:	4, 6, 7, 9, 10, 12, 20, 25, 26, 27, 43
cluelessness:	1, 2, 5, 15, 17, 22
speech:	10, 14, 16, 17, 18, 20, 22-26, 28

Table from (Martinez et al., 2019).

103

Facial expression recognition

casa Paganini infomus

Picture by Dacher Keltner and Daniel T. Cordaro.

104

Approaches

casa Paganini infomus

- Frame-based expression recognition:
  - It does not use temporal information. Rather, it uses information from the current input image with or without a reference frame.
  - Many approaches classify AUs as a preliminary step.
  - Categorical and dimensional models of emotion are employed, leading to classification and regression problems, respectively.
  - Several (mainly machine learning) methods were adopted in the literature, e.g., neural networks, support vector machines, linear discriminant analysis, Bayesian networks, and rule-based classifiers.
  - More recently, deep learning approaches have been used to jointly perform feature extraction and recognition.

105

Approaches

casa Paganini infomus

- Sequence-based expression recognition:
  - It uses the temporal information from a sequence of images to recognize the expressions of one or more frames.
  - Techniques such as hidden Markov models, recurrent neural networks, rule-based classifiers, and sequence-based classifiers were used to deal with temporal information.
  - Most sequence-based recognizers apply categorical models of emotion. More recently, dynamic, continuous models have also been considered, e.g., by applying deep bidirectional long short-term memory recurrent neural networks.

106

Open challenges

- Not just basic emotion, but also face detection of expressions of complex mental states, fatigue, frustration, pain, mood and personality traits, drowsiness, emotional attachment, and indices of psychiatric disorder.
- Micro-expressions: these are brief facial expression people in high stake situations make when trying to conceal feelings. They are subtle and difficult to deal with.
- Reliable ground truth: many datasets are nowadays available for analysis of facial expression, but their manual annotation is a hard task and reliability of labels is typically unknown.
- Robust recognition in naturalistic environments.

107

Open challenges

Robustness	
Rb1	Deal with subjects of different age, gender, ethnicity
Rb2	Handle lighting changes
Rb3	Handle large head motion
Rb4	Handle occlusion
Rb5	Handle different image resolution
Rb6	Recognize all possible expressions
Rb7	Recognize expressions with different intensity
Rb8	Recognize asymmetrical expressions
Rb9	Recognize spontaneous expressions

Automatic process	
Am1	Automatic face acquisition
Am2	Automatic facial feature extraction
Am3	Automatic expression recognition

Real-time process	
Rt1	Real-time face acquisition
Rt2	Real-time facial feature extraction
Rt3	Real-time expression recognition

Autonomic Process	
An1	Output recognition with confidence
An2	Adaptive to different level outputs based on input images

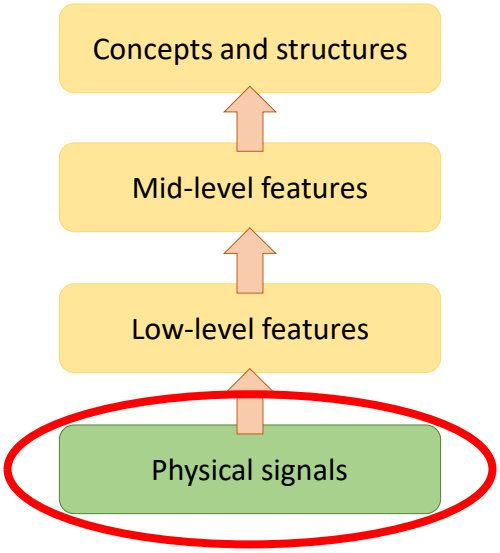
Properties of an ideal facial expression analysis system (Tian et al., 2011)

108

3. Body movement and gesture

1

A conceptual framework

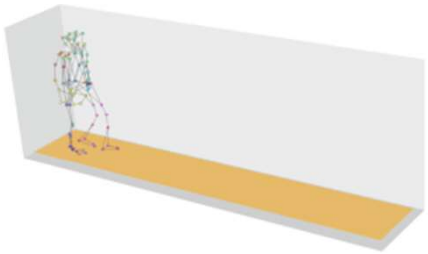


2

Devices for capturing movement

casa Paganini infomus

- Inertial measurement units (IMUs):
  - On-body or handheld devices
  - Magneto-inertial motion capture (MoCap) suites
- Optical devices:
  - Optical motion capture systems
  - Range imaging devices
  - Video cameras
- Other less commonly used devices:
  - Magnetic motion capture systems
  - Mechanical motion capture systems



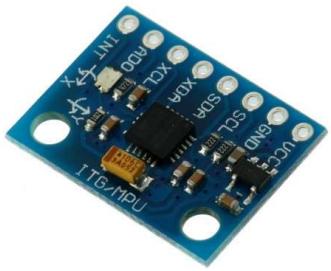
Two repetitions of a walking sequence of an individual recorded using a motion-capture system. Source: Wikipedia. Author: Lars Lau Raket.

3

Inertial measurement units

casa Paganini infomus

- These are electronic devices that measure a body's specific force, angular rate, and sometimes the orientation of the body.
- They use a combination of accelerometers, gyroscopes, and sometimes magnetometers.
- Usually, it is required to wear the sensing devices, or they can be embedded in something we bring with us (e.g., a smartphone).



An example of IMU. Source: https://nicolovaligi.com/ Robotics for developers 6/6: adding an accelerometer

4

Example: on body IMUs

casa Paganini infomus

Source: EU-H2020 ICT Project DANCE. Choreographer and Dancer: Virgilio Sieni.

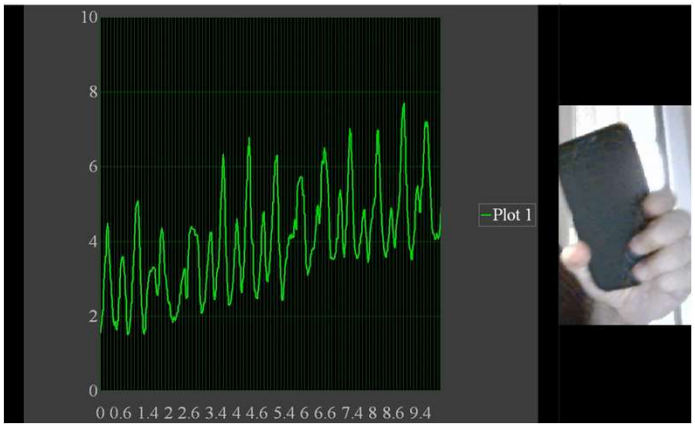
Interactive sonification: movement is captured by means of two IMUs located on the dancer's wrists.

Captured data:
3D components of acceleration

5

Example: handheld devices

casa Paganini infomus



The graph shows the module of acceleration, as computed from the 3 components.

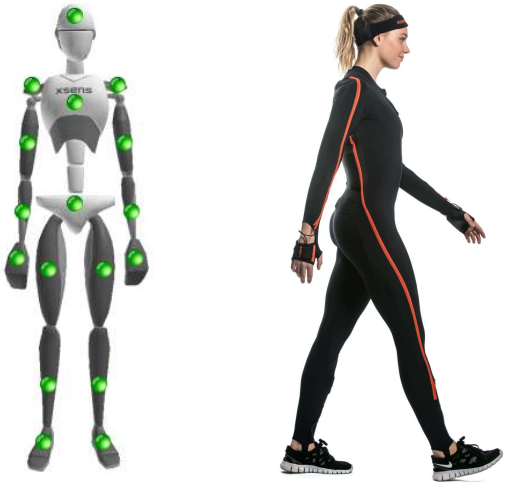
Acceleration data obtained from a smartphone by using the EyesWeb platform.

Captured data:
3D components of acceleration

6

Magneto-inertial suites

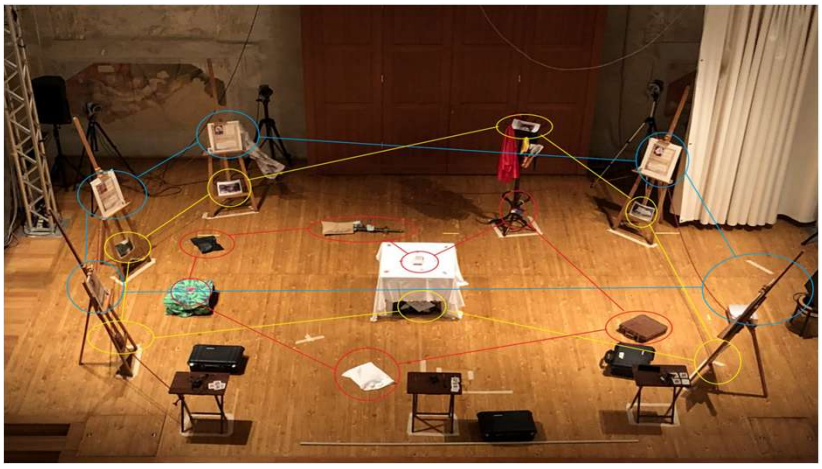
- These are suites that employ multiple IMUs, and possibly include gloves and insole pressure sensors.
- These systems can either capture raw sensor measurements or they can reconstruct 3D skeletons of people moving, also including angular displacements.



An MVN link magneto-inertial suite. Source: Movella, https://base.movella.com

7

Example: magneto-inertial suites



The GAME-ON dataset (Maman et al., 2020)

Captured data: 3D positions rotations

Maman, L., Ceccaldi, E., Lehmann-Willenbrock, N., Likforman-Sulem, L., Chetouani, M., Volpe, G., and Varni, G., 2020. GAME-ON: A Multimodal Dataset for Cohesion and Group Analysis. IEEE Access, 8, 124185-124203.

8

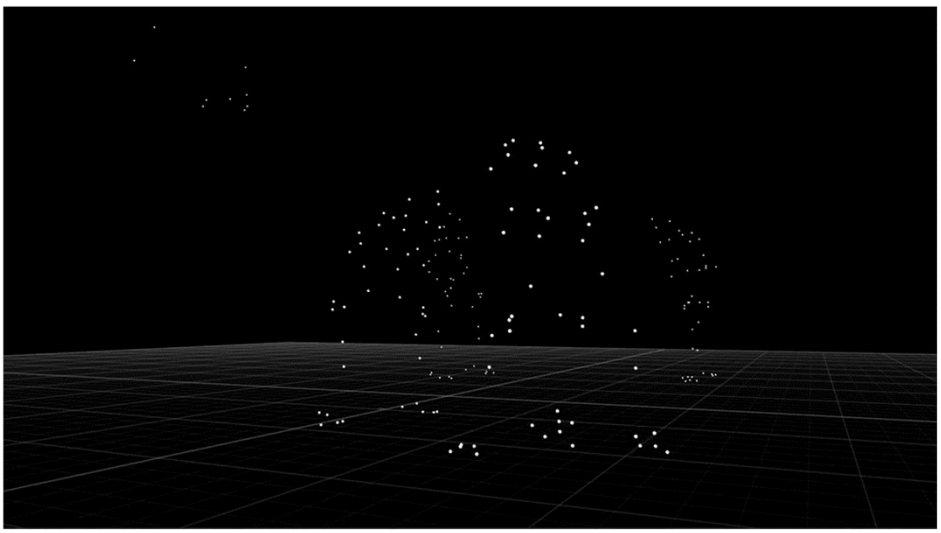
Optical motion capture systems

- They usually require wearing markers and need many video cameras to cover wide areas.
- The captured images are used to triangulate the 3D position of the markers.
- They have high precision and high cost.

A Vicon optical motion capture system. Source: Vicon, https://www.vicon.com/

9

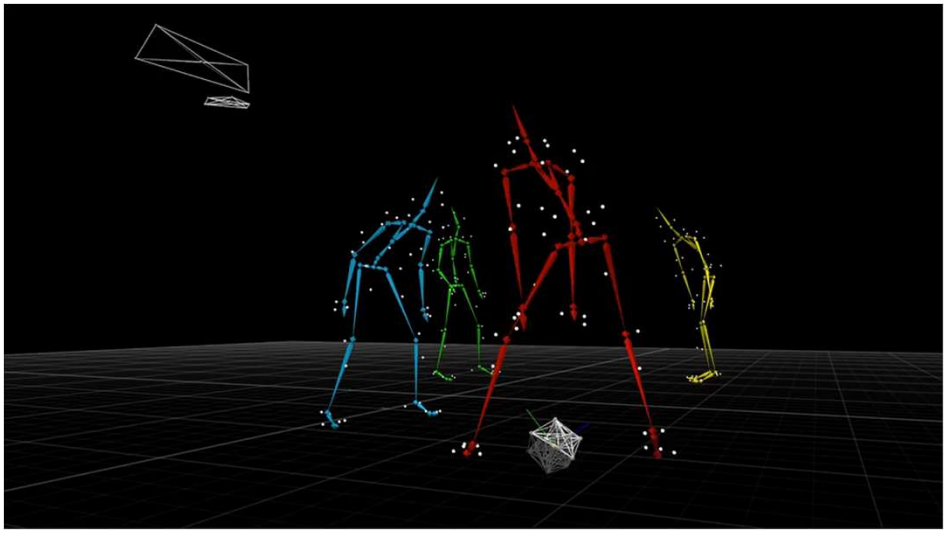
Example: optical MoCap systems



Example of visualization of markers (source: OptiTrack)

10

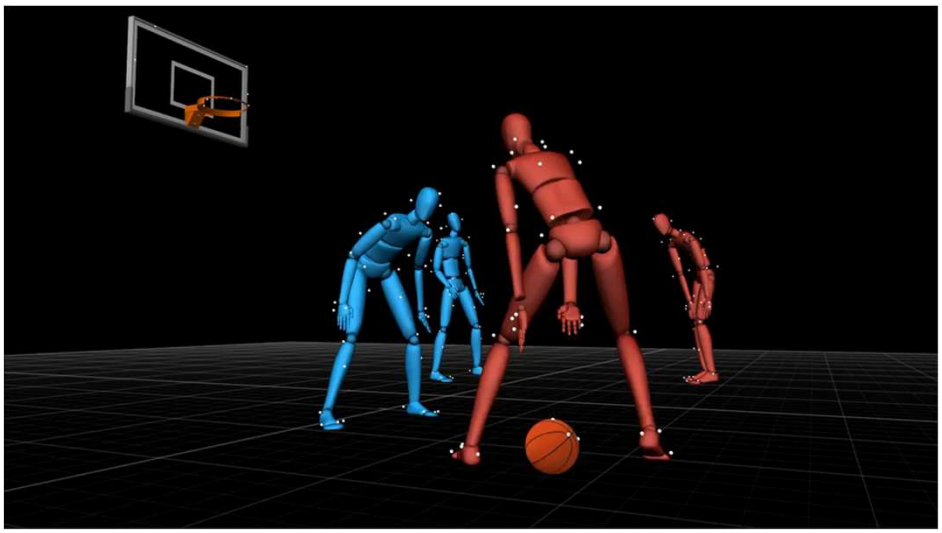
Example: optical MoCap systems



Example of visualization of skeletons (source: OptiTrack)

11

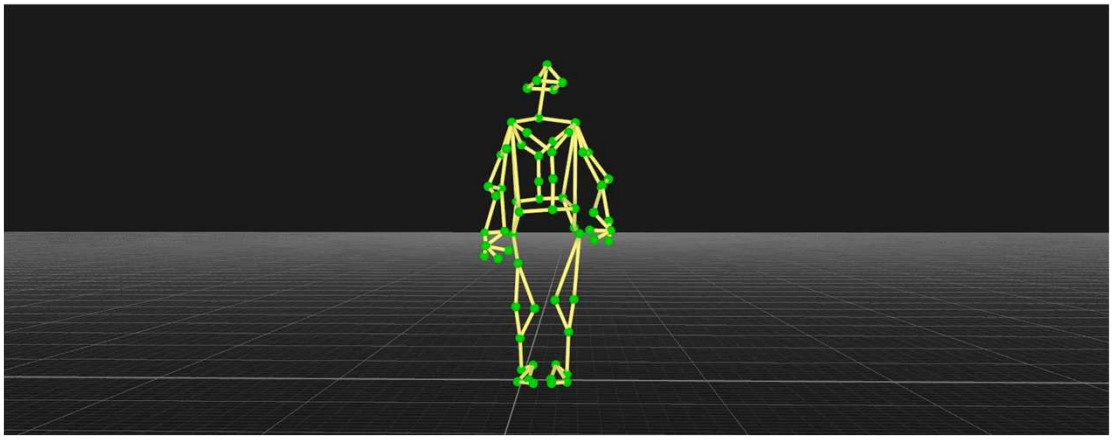
Example: optical MoCap systems



Example of visualization of body models (source: OptiTrack)

12

Example: optical MoCap systems



Interactive sonification, EU-H2020 ICT Project DANCE.
Captured data: 3D positions, orientation of rigid bodies.

13

Range imaging devices

- These are quite cheap devices, but they have a much lower precision than optical MoCap systems.
- They are markerless.

Microsoft Kinect v.2

Intel RealSense

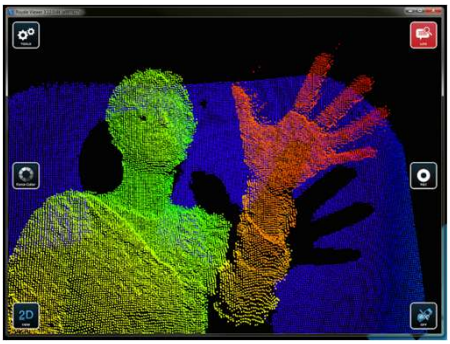
Stereolab ZED 2

14

Range imaging

casa Paganini infomus

- Range imaging is the name for a collection of techniques that produce an image encoding the distance to points in a scene.
- Pixel values of the output image correspond to such distance.
- Different techniques are applied:
  - Stereo triangulation
  - Sheet of light triangulation
  - Structured light
  - Time-of-flight
  - Interferometry
  - Coded Aperture



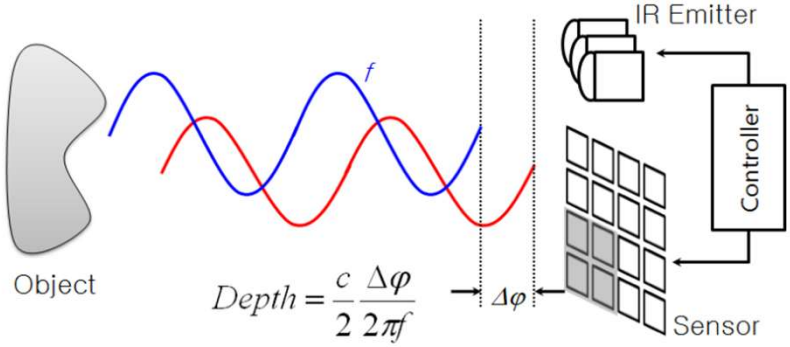
Depth map produced by a pico monstar camera.
Source: https://pmdtec.com/

15

Time of Flight (ToF) Imaging

casa Paganini infomus

- An IR wave is directed to the target object, and the sensor detects the reflected IR component
- Phase difference between radiated and reflected IR waves is computed. Distance to objects is obtained from the difference.



Hansard, M., Lee, S. Choi, O., Horaud, R., 2012. Time of Flight Cameras: Principles, Methods, and Applications. Springer Briefs in Computer Science.

16

Example: range imaging devices

Output of Kinect v.1 (the same is obtained with v.2 and Azure)

Captured data:
3D positions, processed images (depth images, blobs)

17

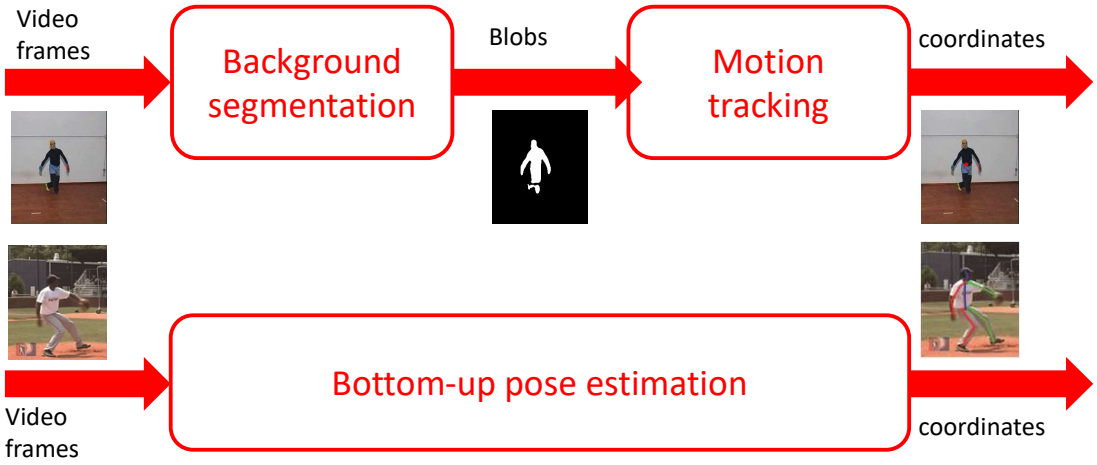
Video cameras

- Ranging from expensive professional cameras to cheap webcams.
- They just capture images.
- So, movement data needs to be extracted from the captured images.

18

Movement data from video

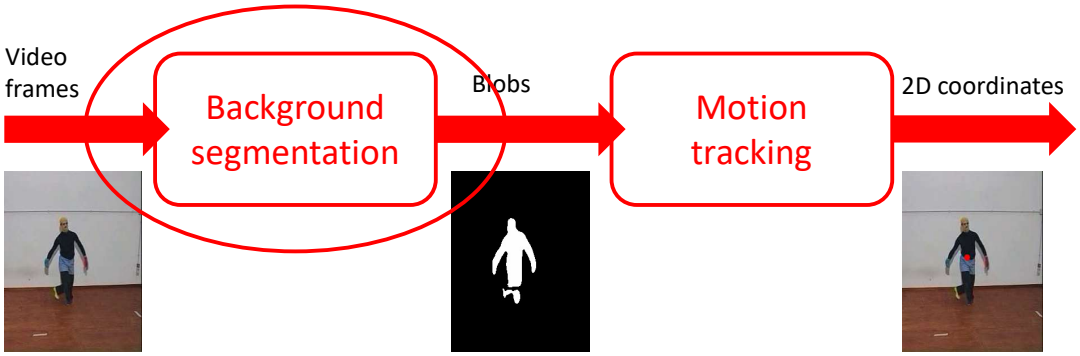
- Approaches:



19

Movement data from video

- Approach 1:
  - Detecting the users (background segmentation).
  - Tracking their movement along time (motion tracking).



20

Background segmentation

casa Paganini infomus

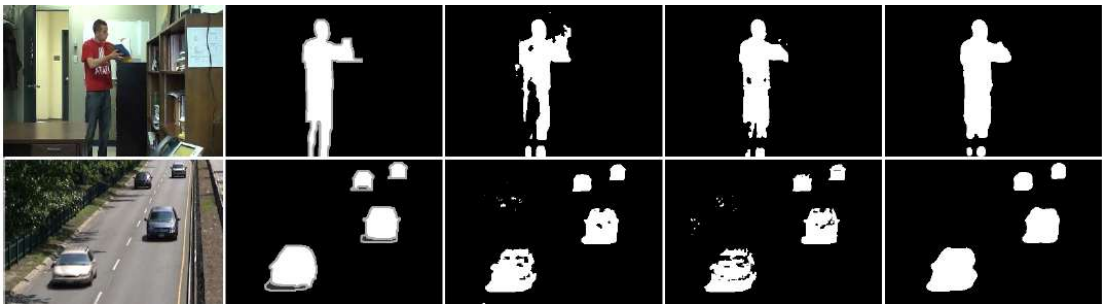
- Background segmentation refers to a collection of techniques for detecting moving objects in a video stream, by separating them from the background.
- It is grounded on the assumption that moving objects (e.g., people, vehicles) are important for interaction.
- Basic approach:
  - Build (and possibly update) a model of the background.
  - Compare the current frame with the model.
  - Separate foreground from background: areas in the current frame that are similar to the model are labeled as background.

21

Background segmentation

casa Paganini infomus

- Two classical examples:
  - Simple background subtraction.
  - Frame differencing.



Background segmentation as performed by different algorithms. Source: St-Charles, P., Bilodeau, G., and Bergevin, R., 2014. Flexible Background Subtraction with Self-Balanced Local Sensitivity. In Proc. 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops, 414-419.

22

Simple background subtraction

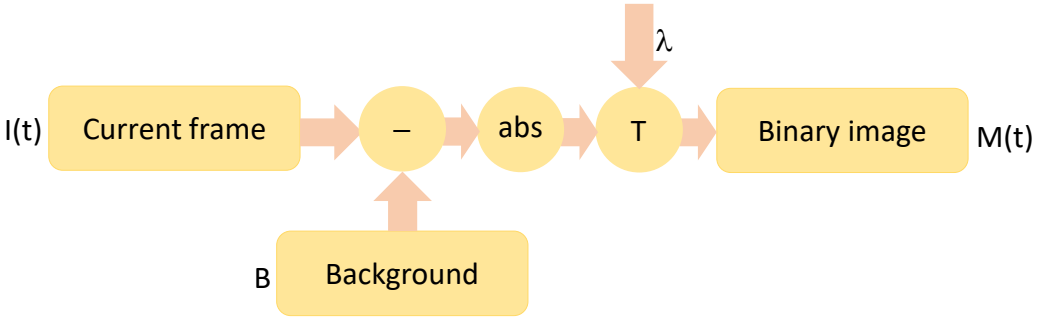
casa Paganini infomus

- The empty background is captured and stored.
- The absolute difference between current image and stored background is computed.
- A threshold is applied to the difference image.
- Pixels having values above the threshold (i.e., difference higher than the threshold) are set to 255 (foreground).
- Pixels having values below the threshold (i.e., difference lower than the threshold) are set to 0 (background).
- The result is a binarized image.

23

Simple background subtraction

casa Paganini infomus




```
B = I(0);
loop time t
  I(t) = next frame;
  diff(t) = abs(B - I(t));
  M(t) = threshold(diff(t),λ);
end
```

24

Simple background subtraction

- Pros:
  - It is easy to implement.
  - It requires low computational power.
  - It works reasonably well in controlled conditions.
- Cons:
  - Pixels in the foreground should have values sufficiently different from those in the background.
  - Objects entering the scene and stopping continue to be detected.
  - If part of the background starts moving, both the moving object and its negative ghost (the revealed background) are detected.

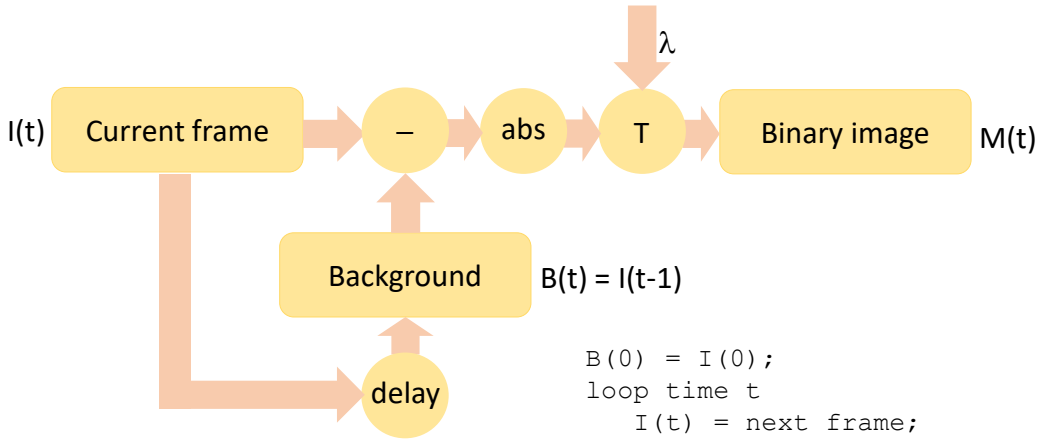
25

Frame differencing

- The background model is the previous image, which is captured and stored.
- The absolute difference between current image and stored background (i.e., the previous image) is computed.
- A threshold is applied to the difference image.
- Pixels having values above the threshold (i.e., difference higher than the threshold) are set to 255 (foreground).
- Pixels having values below the threshold (i.e., difference lower than the threshold) are set to 0 (background).
- The result is a binarized image.

26

Frame differencing




```
B(0) = I(0);
loop time t
   I(t) = next frame;
   diff(t) = abs(B(t) - I(t));
   M(t) = threshold(diff(t),λ);
   B(t) = I(t);
end
```

27

Frame differencing

- Pros:
  - It is easy to implement.
  - It requires low computational power.
  - It quickly adapts to changes.
  - Objects that start moving do not leave ghosts.
- Cons:
  - It only detects leading and trailing edges of uniformly colored objects.
  - Only very few pixels in the foreground are labeled.
  - Objects that stop disappear.
  - It is difficult to detect objects moving towards and away the camera.

28

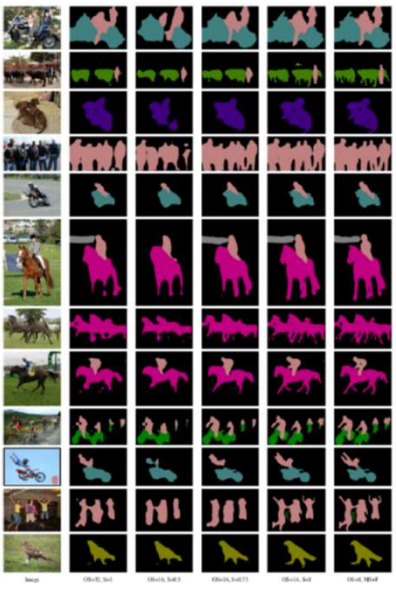
Statistical background modeling

- Background segmentation can be improved, and the drawbacks of the previously mentioned techniques reduced, by computing a statistical model of the background.
- (RGB) values of each pixel in the current image are compared with the model.
- If the values fit with the model, the pixel is labeled as belonging to the background.
- If the values do not fit with the model, the pixel is labeled as belonging to the foreground.
- Example: adaptive mixture of Gaussians (Stauffer and Grimson, 1999)

29

Semantic segmentation

- In computer vision, semantic segmentation is the task that assigns a class label to pixels.
- Background segmentation can be taken as a special case of semantic segmentation.
- Examples of models: DeepLab (Chen et al., 2018), MobileNetV2 (Sandler et al., 2019), and SAM2 (Ravi et al., 2024, developed by Meta).



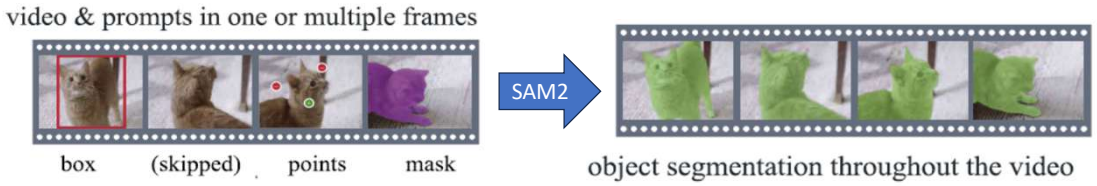
Picture from (Sandler et al., 2019)

30

SAM2: Segment Anything Model 2

casa Paganini infomus

- It extends SAM (Kirillov et al., 2023) to videos.
- Focus on the Promptable Visual Segmentation (PVS) task:
  - Input: points, boxes, or masks annotated on any frame of a video.
  - Output: the predicted spatio-temporal mask (masklet).
  - Once a masklet is predicted, it can be iteratively refined by providing prompts in additional frames.

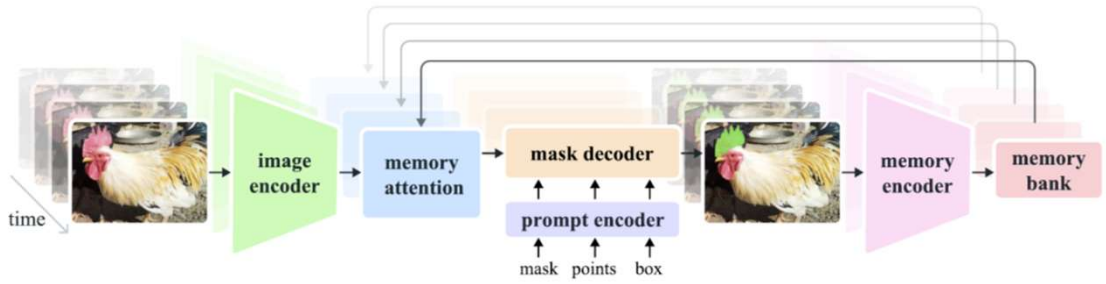


Adapted from: Ravi, N., et al. 2024. Segment Anything in Images and Videos. arXiv:2408.00714.

31

SAM2: architecture

casa Paganini infomus

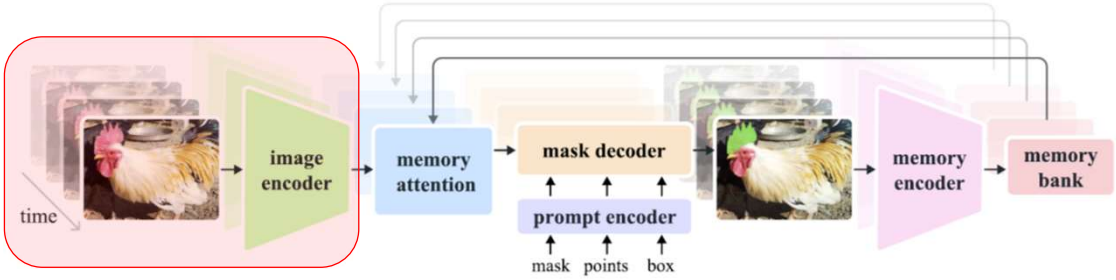


Source: Ravi, N., et al. 2024. Segment Anything in Images and Videos. arXiv:2408.00714.

- Segmentation in the current frame is conditioned on:
  - The current prompt.
  - Previously observed memories.

32

SAM2: architecture

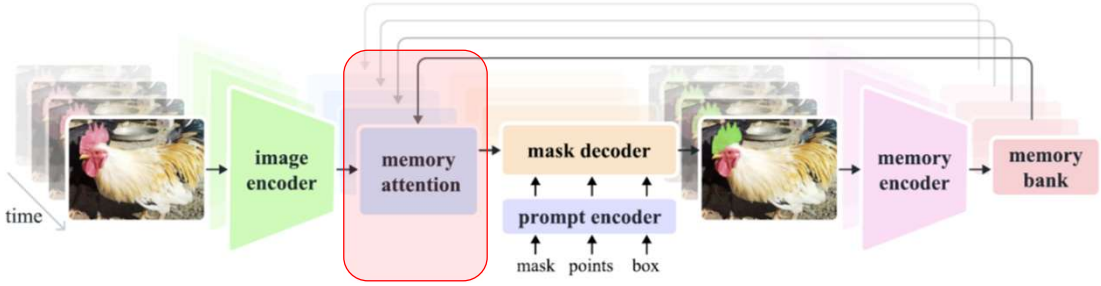


Source: Ravi, N., et al. 2024. Segment Anything in Images and Videos. arXiv:2408.00714.

- Frames are consumed one at a time by the image encoder.
- It combines Hiera (a hierarchical vision transformer, Ryali et al. 2023) with a feature pyramid network (Lin et al., 2017).

33

SAM2: architecture

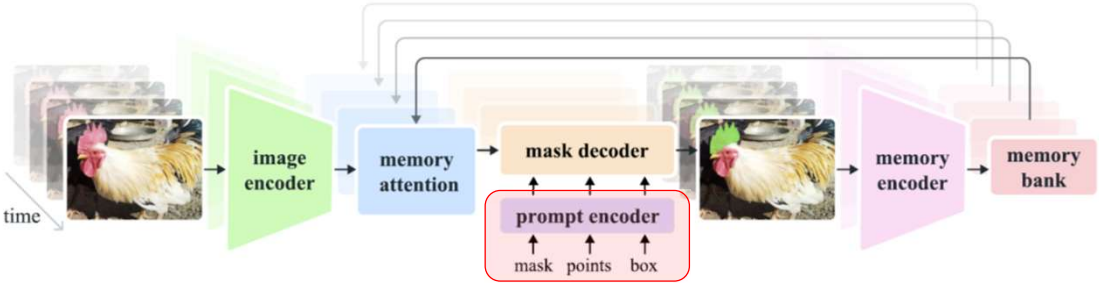


Source: Ravi, N., et al. 2024. Segment Anything in Images and Videos. arXiv:2408.00714.

- The current frame embedding is cross-attended to memories of the target object from previous frames.
- 4 self-attention and cross-attention layers are employed.

34

SAM2: architecture

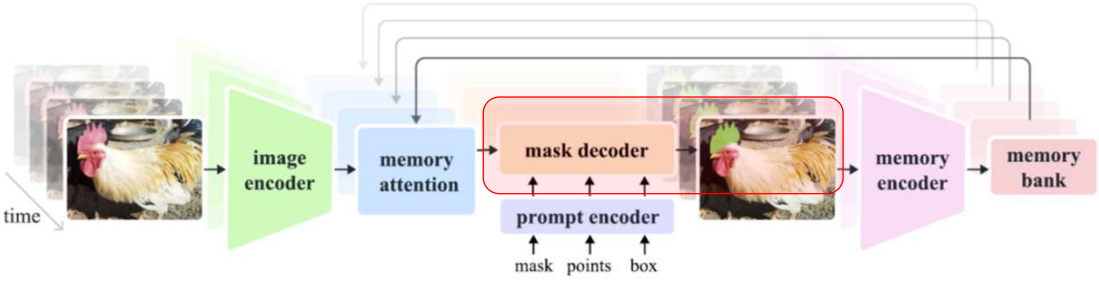


Source: Ravi, N., et al. 2024. Segment Anything in Images and Videos. arXiv:2408.00714.

- Sparse prompts are represented by positional encodings summed with learned embeddings for each prompt type.
- Masks are embedded and summed with frame embedding.

35

SAM2: architecture

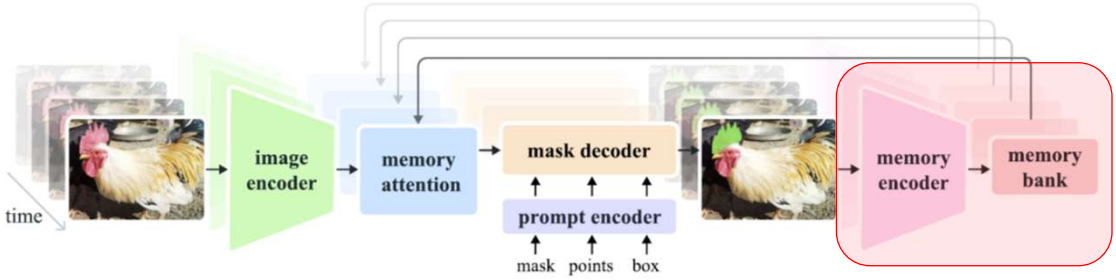


Source: Ravi, N., et al. 2024. Segment Anything in Images and Videos. arXiv:2408.00714.

- A modified transformer decoder block predicts whether the target is present in the frame and its segmentation mask.
- For ambiguous prompts, multiple masks are predicted.

36

SAM2: architecture



Source: Ravi, N., et al. 2024. Segment Anything in Images and Videos. arXiv:2408.00714.

- The output mask goes through a convolutional module. Embeddings are summed element-wise with those from the image encoder. Final embeddings are stored in a FIFO queue of memories of up to N recent frames.

37

SAM2: data collection

- SA-V dataset: 50.9k videos captured by 510 crowd-workers.
- 54% indoor and 46% outdoor scenes, average duration: 14s.
- In-the-wild environments and various everyday scenarios.

	#Videos	Duration	#Masklets	#Masks	#Frames
DAVIS 2017 (Pont-Tuset et al., 2017)	0.2K	0.1 hr	0.4K	27.1K	10.7K
YouTube-VOS (Xu et al., 2018b)	4.5K	5.6 hr	8.6K	197.3K	123.3K
UVO-dense (Wang et al., 2021b)	1.0K	0.9 hr	10.2K	667.1K	68.3K
VOST (Tokmakov et al., 2022)	0.7K	4.2 hr	1.5K	175.0K	75.5K
BURST (Athar et al., 2022)	2.9K	28.9 hr	16.1K	600.2K	195.7K
MOSE (Ding et al., 2023)	2.1K	7.4 hr	5.2K	431.7K	638.8K
Internal	62.9K	281.8 hr	69.6K	5.4M	6.0M
SA-V Manual	50.9K	196.0 hr	190.9K	10.0M	4.2M
SA-V Manual+Auto	50.9K	196.0 hr	642.6K	35.5M	4.2M

Source: Ravi, N., et al. 2024. Segment Anything in Images and Videos. arXiv:2408.00714.

38

SAM2: data annotation

casa Paganini infomus

- Phase 1: Annotators are tasked with annotating the mask of a target object in every frame of a video at 6fps using SAM, and manual editing tools. This approach yields high-quality annotations but is time-consuming (annotation time: 37.8s/ frame). Collected 16k masklets across 1.4k videos.
- Phase 2: Annotators used the same tools as in Phase 1 to generate masks in the first frame and then used SAM2 (trained with data from Phase 1) to temporally propagate the mask. At any subsequent frame, annotators could modify the predictions by annotating a mask from scratch. Collected 63.5k masklets, annotation time: 7.4 s/frame.

39

SAM2: data annotation

casa Paganini infomus

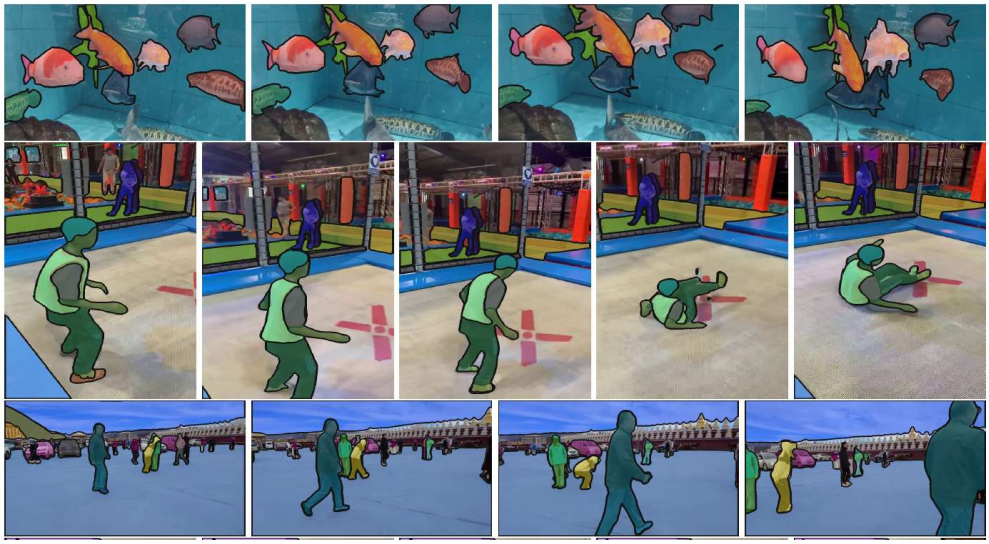
- Phase 3: Annotators provide refinement clicks to the masklets SAM2 predicted (trained with Phase 1 and 2 data). Collected 197k masklets, annotation time: 4.5 s/frame.
- Further automatically generated masklets are obtained by prompting SAM2 with a grid of points in the first frame.
- A separate set of annotators are tasked with verifying the quality of each annotated masklet as satisfactory or not.

	Model in the Loop	Time per Frame	Edited Frames	Clicks per Clicked Frame
Phase 1	SAM only	37.8 s	100.00 %	4.80
Phase 2	SAM + SAM 2 Mask	7.4 s	23.25 %	3.61
Phase 3	**SAM 2**	**4.5 s**	**19.04 %**	**2.68**

Evolution of the annotation process. Source: Ravi, N., et al. 2024.

40

SAM2: data annotation



Examples of annotated frames. Source: Ravi, N., et al. 2024.

41

SAM2: performances

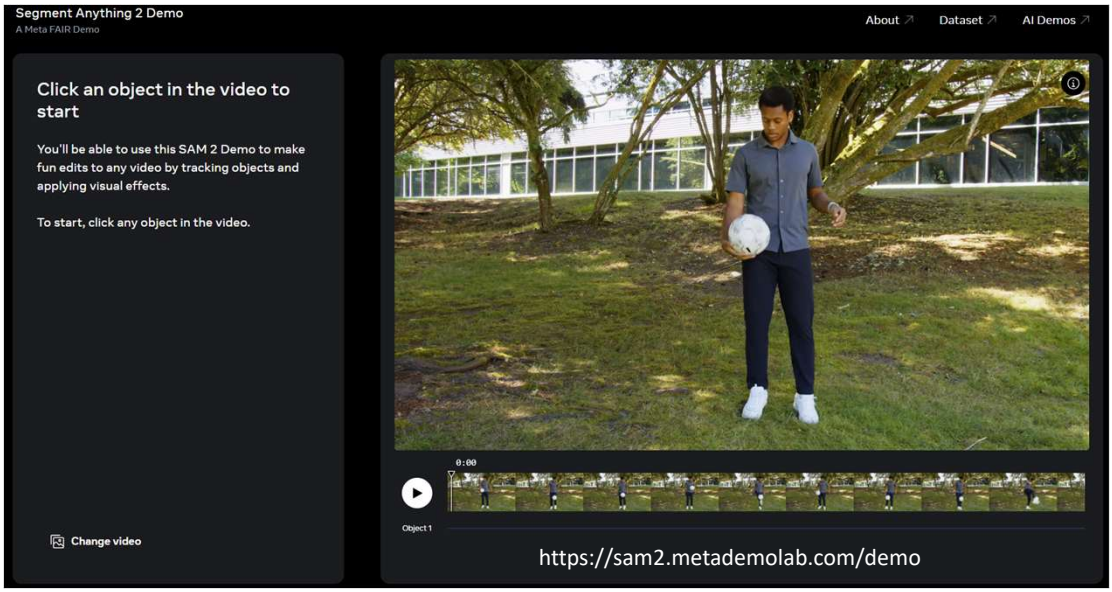
- Baselines: SAM (to provide the mask of the target object in the first frame) followed by XMem++ or Cutie.
- Prompts: 1 click, 3 clicks, or 5 clicks on the first frame.
- Metric: $\mathcal{J}$ & $\mathcal{F}$ accuracy, i.e., the average between region similarity (Jaccard Index, $\mathcal{J}$) and contour accuracy (i.e., the F1-score calculated from the contour points of the segmented mask and the ground truth, $\mathcal{F}$).

Method	1-click	3-click	5-click	bounding box	ground-truth mask‡
SAM+XMem++	56.9	68.4	70.6	67.6	72.7
SAM+Cutie	56.7	70.1	72.2	69.4	74.1
SAM 2	**64.3**	**73.2**	**75.4**	**72.9**	**77.6**

Results of tests performed across 17 datasets. Source: Ravi, N., et al. 2024.

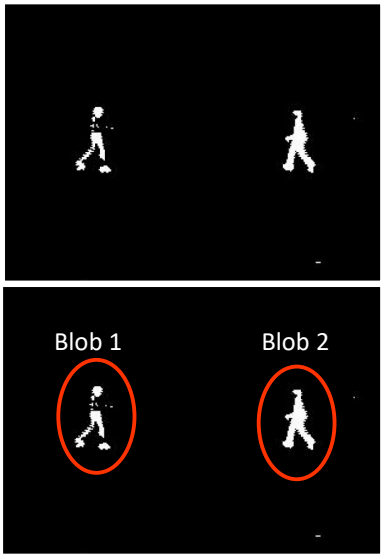
42

SAM2: demo time!



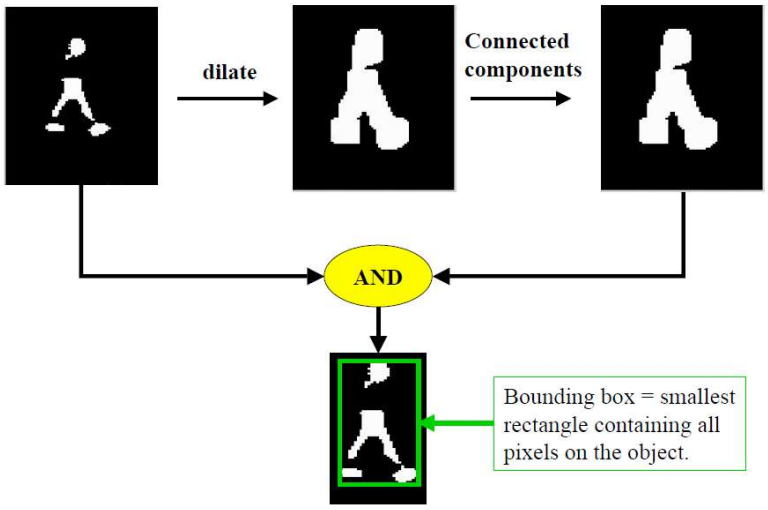
Grouping pixels into blobs

- Background segmentation is a pixel-level process, but higher-level processing is needed.
- Foreground pixels belonging to the same blob need indeed to be grouped together.
- This step is called Connected Component Labeling (CCL).
- Some filtering can be applied both before and after CCL.



Picture by Robert Collins, Pennsylvania State University

Connected Components Labeling

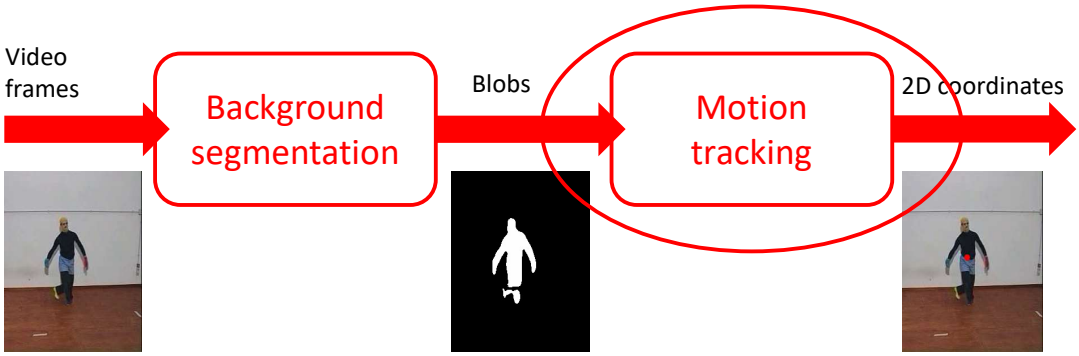


Picture by Robert Collins, Pennsylvania State University

45

Movement data from video

- Approach 1:
  - Detecting the users (background segmentation).
  - Tracking their movement along time (motion tracking).



46

Blob tracking

casa Paganini infomus

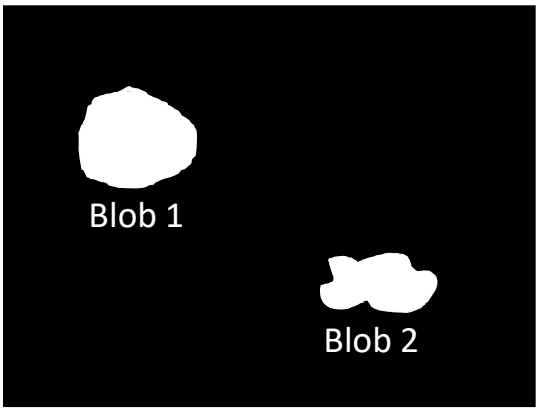
- When an image contains multiple blobs, an important problem is to label them and to follow their movements across frames.
- That is, given a blob and its label at frame *t*, it is needed to identify the same blob (and to give it the same label) at frame *t* + 1.
- This is known as tracking or blob tracking or motion tracking.
- Since gesture is inherently dynamic, i.e., it is a sequence of movements across several frames, blob tracking is of fundamental importance for gesture analysis.

47

Blob tracking

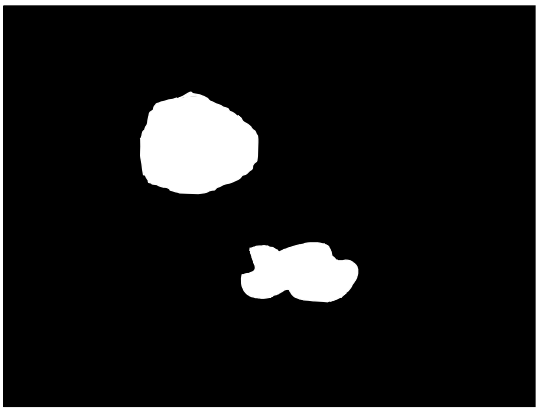
casa Paganini infomus

Frame t



Blobs are labeled

Frame t + 1



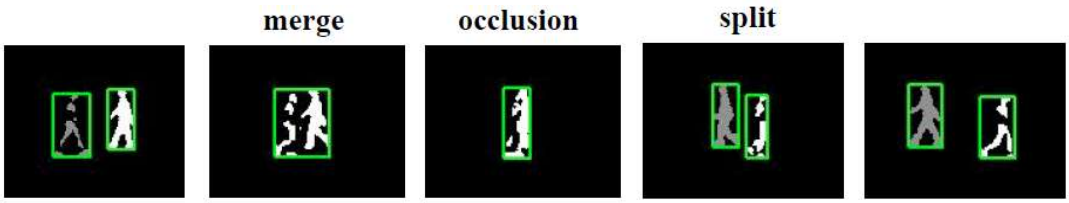
Labels should be assigned:
Which is Blob 1?
Which is Blob 2?

48

Blob tracking

casa Paganini infomus

- Blob tracking has to face many challenging problems, e.g.:
  - When two blobs come close to each other, they merge in a single blob (blob merging).
  - When they separate again (blob splitting), identifying the correct labeling is a challenging problem!
  - When one blob covers another so that the second one is not anymore in view of the video camera, an occlusion is observed.



Pictures by Robert Collins, Pennsylvania State University

49

Blob tracking

casa Paganini infomus

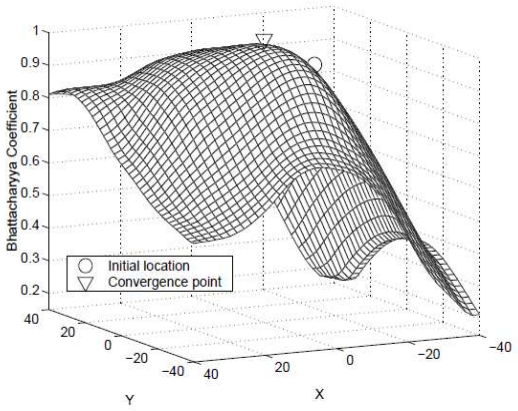
- Blob tracking is based on the assumption that frames are sampled frequently enough, so that features of blobs cannot change too much from a frame to the following one.
- Determining the correspondence of blobs across frames is based on feature similarity between blobs. This is usually done by comparing:
  - Geometrical features, e.g., location, size, shape, overlapping of the bounding box.
  - Kinematic features, e.g., assuming that the tracked objects move with constant velocity.
  - Color histograms.

50

An example: mean-shift

casa Paganini infomus

- Each blob to be tracked is modeled by using a color probability density function.
- Tracking is performed by looking for the maximum similarity between the color probability density function of the model and those of the target candidates.
- Mean shift is a non-parametric technique for locating such a maximum.



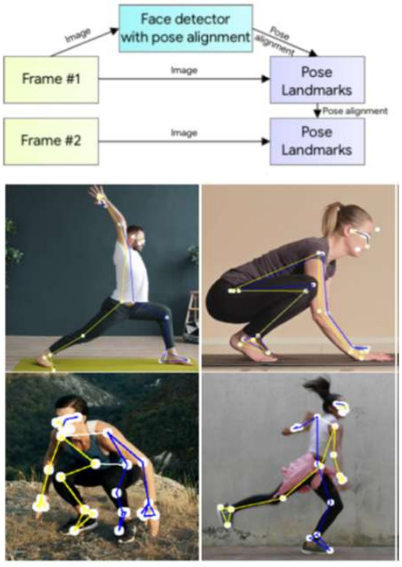
Comaniciu, D., Ramesh, V., Meer, P., 2000. Real-time tracking of non-rigid objects using mean shift. In Proceedings IEEE Conference Computer Vision and Pattern Recognition (CVPR2000), 673-678.

51

An example: Blaze Pose

casa Paganini infomus

- It combines a lightweight body pose detector and a pose tracker network.
- The tracker predicts 33 body landmarks coordinates, the presence of the person on the current frame, and the refined region of interest for the current frame.
- Near real-time on a mobile CPU and can be sped up to super real-time latency on a mobile GPU.



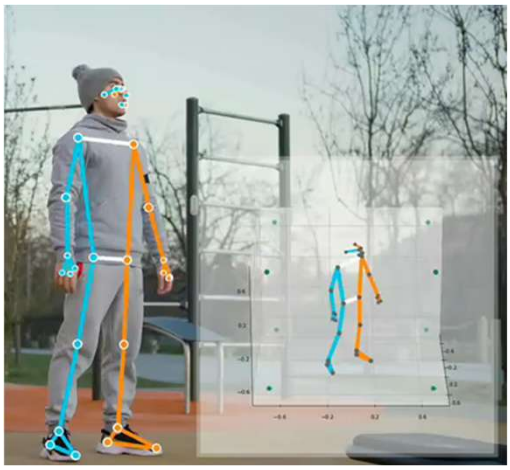
Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., Grundmann, M., 2020. BlazePose: On-device Real-time Body Pose tracking, arXiv:2006.10204.

52

Google MediaPipe Pose

casa Paganini infomus

- It uses a convolutional neural network similar to MobileNetV2 and a variant of BlazePose exploiting a 3D human shape modeling pipeline.
- It achieves real-time performance on most mobile phones, desktops/laptops, in python and even on the web.
- Web demo available at: https://codepen.io/mediapipe

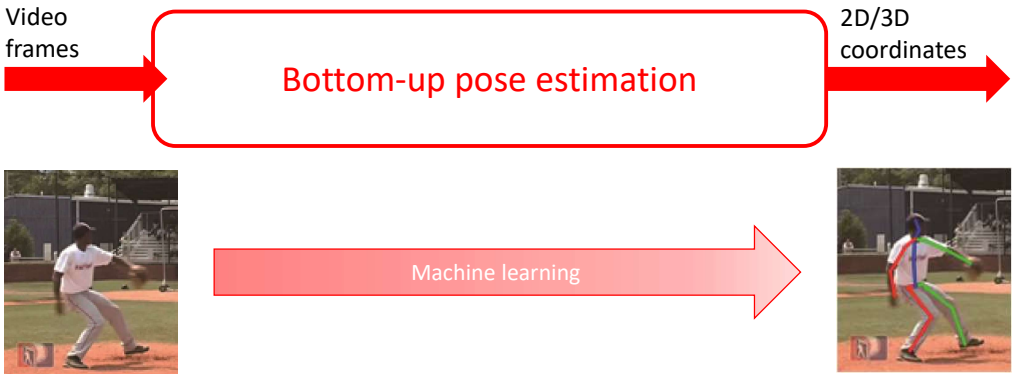


53

Movement data from video

casa Paganini infomus

- Approach 2:
  - Automatic pose estimation leveraging machine learning approaches (e.g., OpenPose, AlphaPose, and so on)



54

An example: OpenPose

casa Paganini infomus

- A bottom-up approach for multi-person pose estimation using a nonparametric representation to learn to associate body parts with individuals in the image (Cao et al., 2016).
- Top-down approaches first detect individuals and then perform single-person pose estimation. Bottom-up approaches reverse this paradigm.

Cao, Z., Simon, T., Wei, S., Sheikh Y., 2016.
Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields,
https://arxiv.org/abs/1611.08050

55

OpenPose

casa Paganini infomus

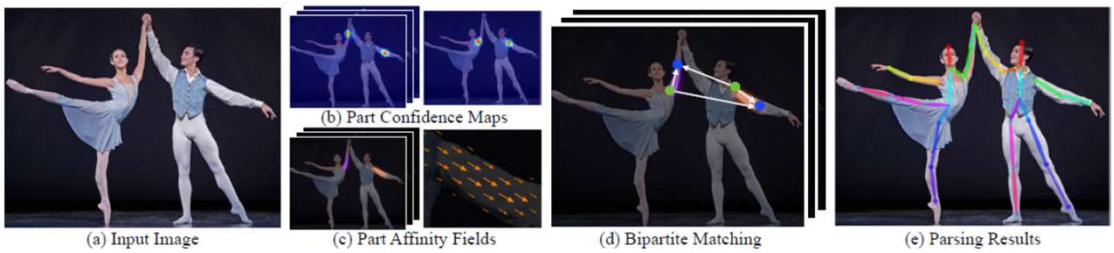
Source: https://colab.research.google.com/github/mmaithani/data-science/blob/main/OpenPose.ipynb

56

OpenPose

casa Paganini infomus

- OpenPose takes the entire image as the input for a Convolutional Neural Network (CNN) to jointly predict:
  - Confidence maps for body part detection
  - Part Affinity Fields (PAFs) for part association.



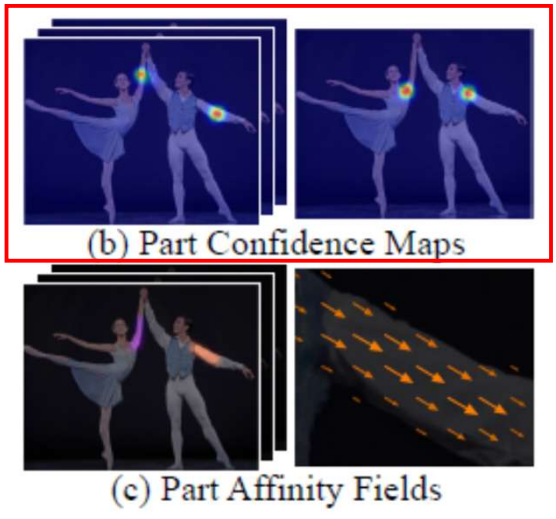
Picture from (Cao et al., 2016): Overall OpenPose pipeline

57

OpenPose

casa Paganini infomus

- Each confidence map is a 2D representation of the belief that a particular body part can be located in any given pixel.
- If a single person appears in the image, a single peak should exist in each confidence map if the corresponding part is visible.
- If multiple people are in the image, there should be a peak corresponding to each visible part for each person.



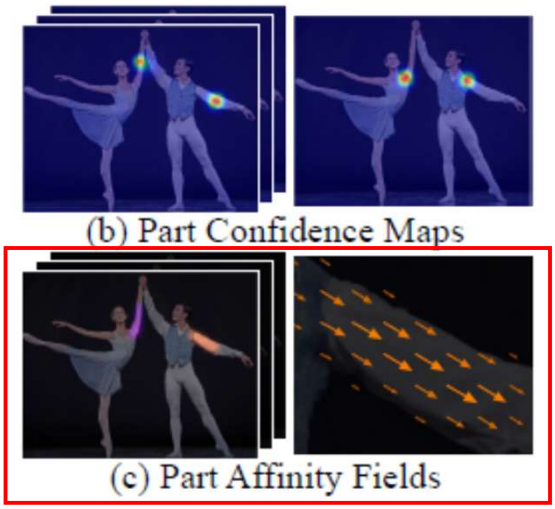
Picture and text from (Cao et al., 2016)

58

OpenPose

casa Paganini infomus

- Part Affinity Fields (PAFs) are a kind of confidence measure that a pair of body parts belong to the same person.
- For each pixel in the area belonging to a particular limb, the corresponding PAF is a 2D vector encoding the direction that points from one part of the limb to the other.



Picture and text from (Cao et al., 2016)

59

OpenPose

casa Paganini infomus

Detected body parts (red and blue dots) for two body part types and all connection candidates (grey lines).

PAFs enable correct association of body parts (yellow arrows).

Picture from (Cao et al., 2016)

60

OpenPose

casa Paganini infomus

- The set of detected body parts is parsed, and the information encoded in PAFs is used to find the pairs of detected parts that are in fact connected limbs, i.e., that belong to the same person.
- This is handled a graph matching problem.

(d) Bipartite Matching

Picture from (Cao et al., 2016)

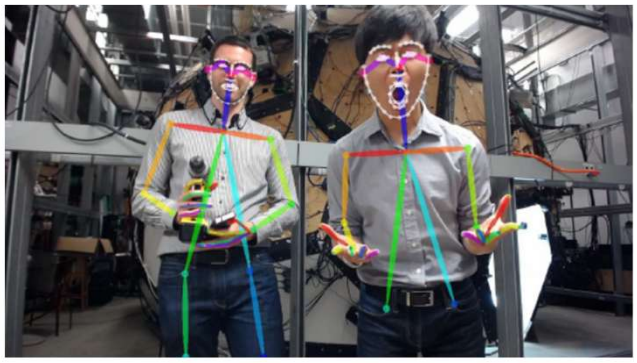
61

OpenPose

casa Paganini infomus

- Parts detected as belonging to the same person are finally connected to yield the output.
- OpenPose can detect body, feet, hands, and facial key-points.

(e) Parsing Results



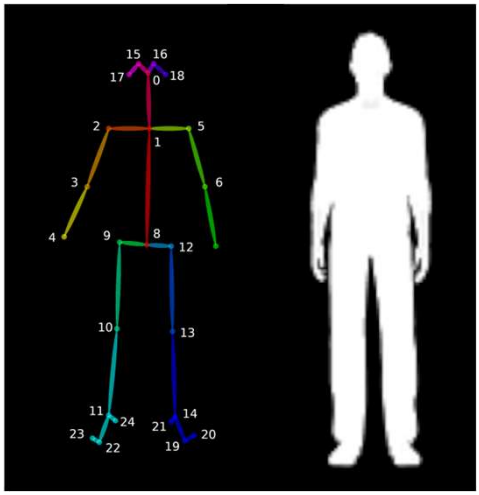
Pictures from (Cao et al., 2016)

62

The output of layer 1…

casa Paganini infomus

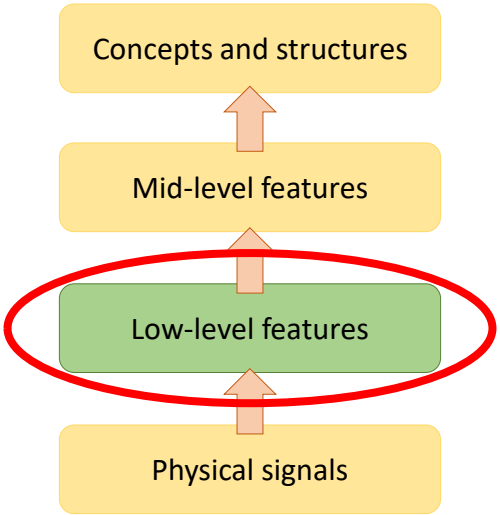
- Summarizing, depending on the used sensor devices and on the applied techniques, this can be one or more of the following:
  - 3D or 2D positions of landmarks.
  - Angles representing rotations.
  - Blobs.
- This is what layer 2 receives as input.



63

A conceptual framework

casa Paganini infomus



64

Input: movement representations

casa Paganini infomus

- Movement $\boldsymbol{X}$ can be represented as a time-series of poses:

$$\boldsymbol{X} = \{\boldsymbol{x}(t_i)\}_{i=1\ldots N} = (\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_2), \ldots, \boldsymbol{x}(t_N))$$

  where $\boldsymbol{x}(t_i)$ is the pose at time t_i. N is the number of poses.

- Movement conventionally starts at time t_0.
- $\Delta t = t_i - t_{i-1}$ is the sampling period, which depends on the adopted sensor devices and is usually constant.
- $f_s = 1/\Delta t$ is the corresponding sampling frequency. Typical values for f_s are 100Hz, 50Hz, 30Hz, 25Hz.

65

Input: movement representations

casa Paganini infomus

- Pose $\boldsymbol{x}(t_i)$ may consist of:
  - A set of M_p positions (e.g., joint positions)

    $\mathrm{P}(t_i) = \{\boldsymbol{p}^1(t_i), \boldsymbol{p}^2(t_i), \ldots, \boldsymbol{p}^{M_p}(t_i)\} \quad \boldsymbol{p}^k = [x^k, y^k, z^k]^{\mathrm{T}} \in \mathbb{R}^3$

    and/or

  - A set of M_q angles (e.g., joint angles), represented as quaternions:

    $\mathrm{Q}(t_i) = \{\boldsymbol{q}^1(t_i), \boldsymbol{q}^2(t_i), \ldots, \boldsymbol{q}^{M_q}(t_i)\} \quad \boldsymbol{q}^h \in \mathbb{R}^4$

- In case both positions and angles are available, then $\boldsymbol{x} = [\boldsymbol{p}^1, \boldsymbol{p}^2, \ldots, \boldsymbol{p}^{M_y}, \boldsymbol{q}^1, \boldsymbol{q}^2, \ldots, \boldsymbol{q}^{M_q}]$ i.e., $\boldsymbol{x} \in \mathbb{R}^{3M_p+4M_q}$.

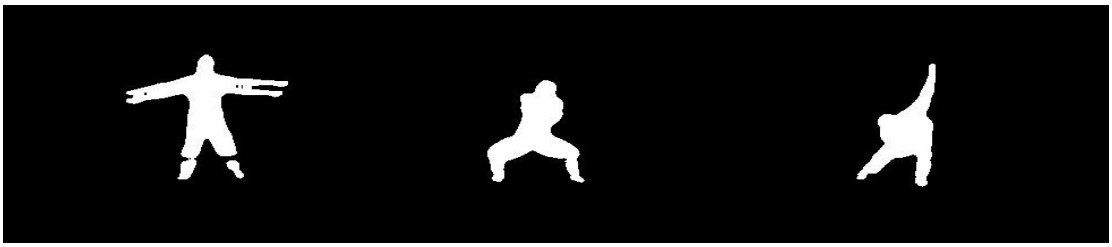
66

Input: movement representations

casa Paganini infomus

- Alternatively, $\boldsymbol{x}(t_i)$ may consist of a blob, i.e., $\boldsymbol{x}(t_i) = \mathrm{B}(t_i)$.
- In such a case, movement $\boldsymbol{X}$ is represented as a time-series of blobs, i.e., by the temporal sequence of the blobs having the same label at times t_0 … t_N:

$$\boldsymbol{X} = \mathrm{B} = (\mathrm{B}(t_0), \mathrm{B}(t_1), \mathrm{B}(t_2), \ldots, \mathrm{B}(t_N))$$

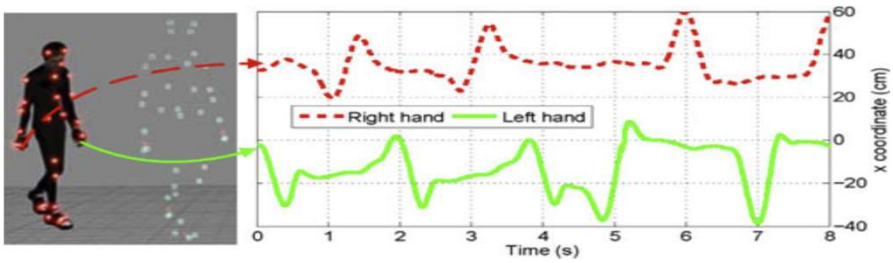


67

Input: movement representations

casa Paganini infomus

- Different capture devices and analysis techniques produce all or part of these representations:
  - *Inertial motion capture systems* usually provide 3D positions and rotations of sensors, i.e., $\boldsymbol{x} = [\boldsymbol{p}^1, \boldsymbol{p}^2, \ldots, \boldsymbol{p}^{M_I}, \boldsymbol{q}^1, \boldsymbol{q}^2, \ldots, \boldsymbol{q}^{M_I}]$.
  - *Optical motion capture systems* usually provide 3D positions of markers, i.e., $\boldsymbol{x} = [\boldsymbol{p}^1, \ldots, \boldsymbol{p}^{M_O}]$.



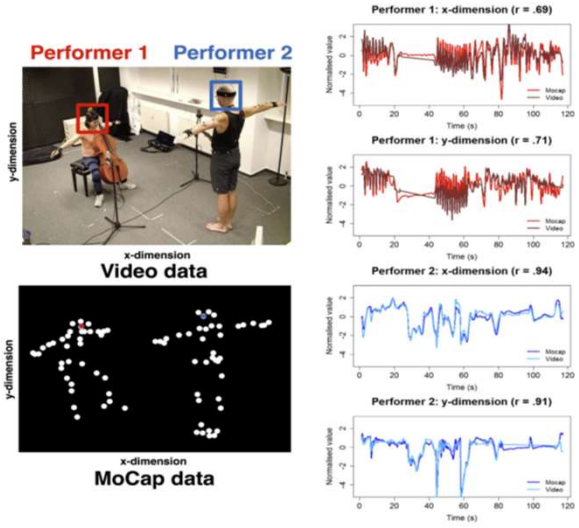
Source: Hou, J., Chau, L., Magnenat-Thalmann, N., and He, Y., 2015. Human Motion Capture Data Tailored Transform Coding. IEEE Transactions on Visualization and Computer Graphics, 21(7), 848-859.©

68

Input: movement representations

casa Paganini infomus

- *RGB-D devices* provide 3D positions of body landmarks. They may provide blobs. That is: $\boldsymbol{x} = [\boldsymbol{p}^1, \dots, \boldsymbol{p}^{M_R}, \mathrm{B}]$.
- *Pose estimation techniques* provide 2D or 3D positions of body landmarks. That is: $\boldsymbol{x} = [\boldsymbol{p}^1, \dots, \boldsymbol{p}^{M_E}]$.
- *Background subtraction and motion tracking* applied to images from a video camera provide a sequence of blobs, i.e., $\boldsymbol{x} = \mathrm{B}$.



Source: Jakubowski, K., Eerola, T., Alborno, P., Volpe, G., Camurri, A., and Clayton, M., 2017. Extracting Coarse Body Movements from Video in Music Performance: A Comparison of Automated Computer Vision Techniques with Motion Capture Data. Frontiers in Digital Humanities, 4:9.

69

Output: features

casa Paganini infomus

- Given a movement $\boldsymbol{X} = (\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \boldsymbol{x}(t_2), \dots, \boldsymbol{x}(t_\mathrm{N}))$, received as input, the output of layer 2 is a set of T time-series $\mathrm{F} = \{\mathrm{F}_1, \mathrm{F}_2 \dots \mathrm{F}_\mathrm{T}\}$ characterizing movement over time.
- Time-series $\mathrm{F}_k = \left(f_k(\boldsymbol{x}(t_0)), f_k(\boldsymbol{x}(t_1)), \dots, f_k(\boldsymbol{x}(t_\mathrm{N}))\right)$ is thus a feature that describes a specific facet of movement.
- Computation depends on data available in $\boldsymbol{X}$, i.e., different features are computed on positions, quaternions, and blobs.
- Features come from physics, biomechanics, psychology (e.g., Boone and Cunningham, 1998; de Meijer, 1989; Walbott 1998), and research in arts and humanities (e.g., Laban).

70

Output: features

casa Paganini infomus

- Example:

Descriptions of Specific Body Features and How They May Be Attributed to the Recognition of Specific Affective States

Source: Kleinsmith, A., and Bianchi-Berthouze, N., 2013. Affective Body Expression Perception and Recognition: A Survey. IEEE Transactions on Affective Computing, 4(1), 15-33.

Aff states/dims	Study	Discriminating features
Anger	Coulson [100]	Head bent back, no backward chest bend, no abdominal twist, arms raised forward & upward
	De Meijer [101]	Bowed trunk & head, knees slightly bent, slow velocity, strong force, downward body movement, stepping backward, arms open frontally
	Dahl & Friberg [96]	Both instruments: Large, very jerky, somewhat fast movements
	Gross et al [27]	High energy, expanded limbs, tense and controlled flow
	Kleinsmith et al [80]	Head bent forward, elbows bent and laterally extended
	Roether et al [91]	Head bent forward, elbows bent
Cold anger	Wallbott [40]	Lateralized hand/arm movements, arms stretched out frontal
Hot anger	Wallbott [40]	Shoulders lifted, lateralized hand/arm movements, arms stretched out frontal
Anxious	Gross et al [27]	Low energy, slow movement, somewhat expanded limbs and torso
Arousal	Kleinsmith et al [103]	Low-arousal: head bent forward, hands close to the body. High arousal: head bent backward, hands vertically extended
Avoidance	Kleinsmith et al. [103]	Vertical extension and lateral opening of the body for high avoidance
Boredom	Wallbott [40]	Collapsed upper body, head bent backwards
Contempt	De Meijer [101]	Bowed trunk & head, knees slightly bent, stepping backward
Concentrating	Kleinsmith et al [60]	Shoulders slumped forward, the arms extended down and diagonally across the body
Defeated	Kleinsmith et al [60]	Shoulders slumped forward, the arms extended down and diagonally across the body
Despair	Wallbott [40]	Shoulders forward
Disgust	De Meijer [101]	Bowied trunk & head, knees slightly bent
	Wallbott [40]	Shoulders forward, head bent forward, arms crossed in front
Fear	Coulson [100]	Backward head bend, no abdominal twist, forearms raised, weight shift backward
	De Meijer [101]	Bowed trunk & head, knees slightly bent, downward, backward fast movement, muscles tensed
	Dahl & Friberg [96]	Saxophone: regular, smooth & slow movements. Bassoon: jerky & somewhat fast movements
	Kleinsmith et al [100]	Head straight up or bent back slightly, elbows bent, arms lateral
	Roether et al [91]	Head upright, elbows bent
	Wallbott [40]	Shoulders forward
Frustrated	Kleinsmith et al [60]	Shoulders straight up or back, arms raised and extended laterally
Grief	De Meijer [101]	Bowed trunk & head, knees slightly bent, slow velocity, downward body movement, arms folded across chest
Content	Gross et al [27]	Expanded limbs and torso, low energy

71

Output: features

casa Paganini infomus

- Example (continued):

Descriptions of Specific Body Features and How They May Be Attributed to the Recognition of Specific Affective States

Source: Kleinsmith, A., and Bianchi-Berthouze, N., 2013. Affective Body Expression Perception and Recognition: A Survey. IEEE Transactions on Affective Computing, 4(1), 15-33.

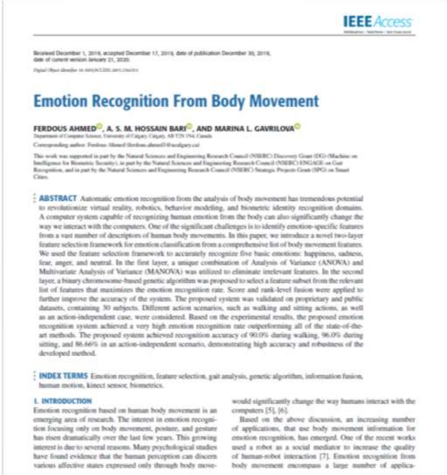
Happiness	Coulson [100]	Head bent back, no forward chest movement, arms raised above shoulder, straight at elbow
	Dahl & Friberg [96]	Saxophone: Large, regular, fluid, somewhat slow movements. Bassoon: Large, fairly regular, jerky, fast movements
	De Silva et al. [102]	Vertical and lateral extension of the arms, opening of the shoulers
	Kleinsmith et al [80]	Head bent back, elbows bent, arms raised
	Roether et al [91]	Head upright, straight spine, arms straight
Joy	De Meijer [101]	Straight trunk & legs, upward, forward, fast body movement, muscles tensed, arms open frontally
	Gross et al [27]	Expanded limbs and torso
Elated joy	Wallbott [40]	Shoulders lifted, head bent backwards, arms stretched frontal
Interest	De Meijer [101]	Straight trunk & legs, stepping forward, muscles relaxed, slow velocity, arms open frontally
Potency	Kleinsmith et al. [103]	Low potency: hands along body. High potency: hands raised to shoulder & extended frontally
Pride	Wallbott [40]	Head bent backwards, arms crossed in front
	Gross et al [27]	Expanded limbs and torso, high energy, hurried, tense and controlled flow
Sadness	Coulson [100]	Forward head bend, forward chest bend, no abdominal twist, arms at side of the trunk
	Dahl & Friberg [96]	Small, very slow, very fluid, fairly regular movements; Bassoon: very little movement
	Castellano et al [97]	Low level of upper body movement, slow velocity of head movements
	Gross et al [27]	Low energy, slow, tense and controlled flow
	De Silva et al. [102]	Little to no vertical or lateral extension of the arms
	Kleinsmith et al [80]	Head bent forward, arms extended straight down alongside body
	Roether et al [91]	Head bent forward, arms straight
	Wallbott [40]	Collapsed upper body
Serene	Castellano et al [97]	High velocity of head movements, high quality of motion
Shame	De Meijer [101]	Bowed trunk & head, knees slightly bent, downward body movement, light force (muscles relaxed), slow velocity, stepping backward
	Wallbott [40]	Collapsed upper body
Surprise	Coulson [100]	Backward head & chest bends, abdominal twisting, arms raised with forearms straight
	De Meijer [101]	Straight trunk and legs, backward stepping, fast velocity
Terror	Wallbott [40]	Arms stretched sideways
Threatening	Aronoff et al [92]	Diagonality & angularity of both arms & movement, diagonal poses
Triumphant	Kleinsmith et al [60]	Shoulders straight up or back, arms raised and extended laterally
Valence	Kleinsmith et al. [103]	High valence has vertical extension of the arms & greater 3D distance between heels
Warmth	Aronoff et al [92]	Roundedness of both arms and body movement, more static and moving arabesques
Admiration	De Meijer [101]	Straight trunk & legs, upward body movement, stepping forward, arms open frontally
Antipathy	De Meijer [101]	Bowed trunk & head, knees slightly bent, stepping backward
Sympathy	De Meijer [101]	Straight trunk and legs, stepping forward, arms open frontally, muscles relaxed, slow velocity

72

Output: features

casaPaganini infomus

- Following a throughout review of literature on the features used for emotion recognition from body movement, Ahmed and colleagues (2020) identified 10 groups of movement features that can be computed from the 3D trajectories of body landmarks.


IEEE Access

Emotion Recognition From Body Movement

FERDOUS AHMED, A. S. M. HOSSAIN BARI, AND MARINA L. GAVRILOVA

ABSTRACT Automatic emotion recognition from the analysis of body movement has tremendous potential to revolutionize virtual reality, robotics, behavior modeling, and biometric identity recognition domains. A computer system capable of recognizing human emotion from the body can also significantly change the way we interact with the computers. One of the significant challenges is to identify emotion-specific features from a vast number of descriptors of human body movements. In this paper, we introduce a novel two-layer feature selection framework for emotion classification from a comprehensive list of body movement features. We used the feature selection framework to accurately recognize five basic emotions: happiness, sadness, fear, anger, and neutral. In the first layer, a unique combination of Analysis of Variance (ANOVA) and Multivariate Analysis of Variance (MANOVA) was utilized to eliminate irrelevant features. In the second layer, a binary chromosome-based genetic algorithm was proposed to select a feature subset from the relevant list of features that maximizes the emotion recognition rate. Score and rank-level fusion were applied to further improve the accuracy of the system. The proposed system was validated on proprietary and public datasets, containing 30 subjects. Different action scenarios, such as walking and sitting actions, as well as an action-independent case, were considered. Based on the experimental results, the proposed emotion recognition system achieved a very high emotion recognition rate outperforming all of the state-of-the-art methods. The proposed system achieved recognition accuracy of 90.0% during walking, 96.0% during sitting, and 86.66% in an action-independent scenario, demonstrating high accuracy and robustness of the developed method.

INDEX TERMS Emotion recognition, feature selection, gait analysis, genetic algorithm, information fusion, human motion, kinect sensor, biometrics.

I. INTRODUCTION

Emotion recognition based on human body movement is an emerging area of research. The interest in emotion recognition focusing only on body movement, posture, and gesture has risen dramatically over the last few years. This growing interest is due to several reasons. Many psychological studies have found evidence that the human perception can discern various affective states expressed only through body move- would significantly change the way humans interact with the computers [5], [6].

Based on the above discussion, an increasing number of applications, that use body movement information for emotion recognition, has emerged. One of the recent works used a robot as a social mediator to increase the quality of human-robot interaction [7]. Emotion recognition from body movement encompass a large number of applica-

Ahmed, F., Bari, A. S. M. H., and Gavrilova, M. L., 2020. Emotion Recognition From Body Movement. IEEE Access, 8, 11761-11781

73

Movement trajectories

casaPaganini infomus

- We assume that pose $\boldsymbol{x}(t_i)$ includes information about the coordinates $\mathrm{P}(t_i) = \{\boldsymbol{p}^1(t_i), \boldsymbol{p}^2(t_i), \dots, \boldsymbol{p}^{M_p}(t_i)\}$ of a set of M_p body landmarks (e.g., body joints).
- The trajectory of the k-th point in $\mathbf{X} = (\boldsymbol{x}(t_0), \boldsymbol{x}(t_1), \dots, \boldsymbol{x}(t_N))$ is given by the time-series:

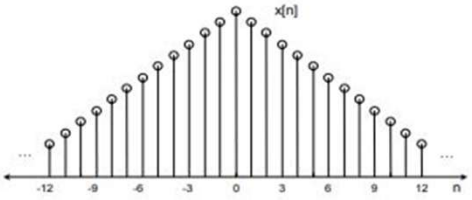
$$\mathbf{P}^k = (\boldsymbol{p}^k(t_0), \boldsymbol{p}^k(t_1), \dots, \boldsymbol{p}^k(t_N))$$

74

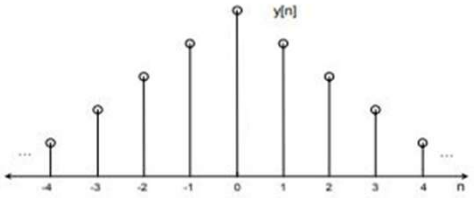
Pre-processing

casa Paganini infomus

- Before extracting features from trajectories, these are usually pre-processed to prepare data.
- Down-sampling: in case the sampling frequency is such that data is too fine-grained for the operations to be performed, trajectories are appropriately down-sampled (e.g., from 100Hz to 50Hz) to reduce computational cost.



Orignal Discrete Signal



Decimation of a discrete-time signal by a factor of 3.

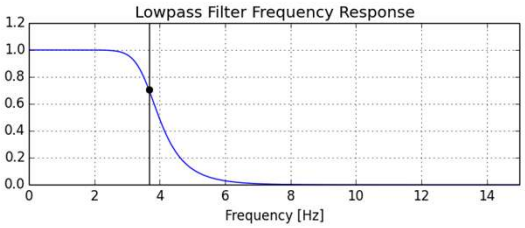
Source: https://www.geeksforgeeks.org/down-sampling-in-matlab/

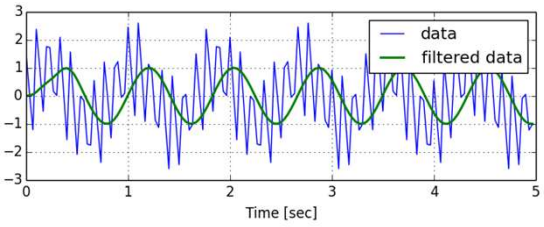
75

Pre-processing

casa Paganini infomus

- Low-pass filtering: this is done to clean the data from noise.
- Skogstad and colleagues (2013) proposed specific low-pass filters for motion capture data with different cut-off frequencies depending on the amount of noise to filter out.



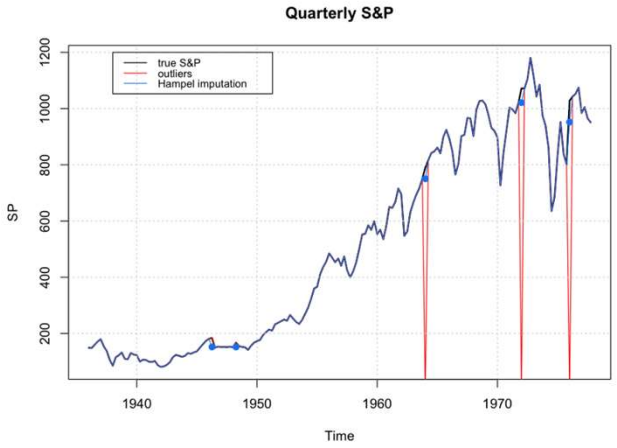


Source: https://stackoverflow.com/questions/25191620/creating-lowpass-filter-in-scipy-understanding-methods-and-units

76

Pre-processing

- Removal of outliers: sample values which are unrealistic are removed from the data (e.g., by using a Hampel filter).



Example: application of a Hampel filter to the S&P composite index from 1936 Q1 to 1977 Q4

Source: medium.com/wwblog/clean-up-your-time-series-data-with-a-hampel-filter-58b0bb3ebb04. Author: Willie Wheeler

77

Group 1: kinematics

- Velocity is the rate of change of the position of a landmark:

$$\boldsymbol{v}^k(t_i) = \frac{\boldsymbol{p}^k(t_i) - \boldsymbol{p}^k(t_{i-1})}{t_i - t_{i-1}}$$

- Centered derivative can be used to reduce noise:

$$\boldsymbol{v}^k(t_{i-1}) = \frac{\boldsymbol{p}^k(t_i) - \boldsymbol{p}^k(t_{i-2})}{t_i - t_{i-2}}$$

- Speed is defined as the magnitude of velocity:

$$v^k(t_i) = \sqrt{\left(v_x^k(t_i)\right)^2 + \left(v_y^k(t_i)\right)^2 + \left(v_z^k(t_i)\right)^2}$$

78

Group 1: kinematics

casa Paganini infomus

- Acceleration is the rate of change of velocity:

$$\boldsymbol{a}^k(t_i) = \frac{\boldsymbol{v}^k(t_i) - \boldsymbol{v}^k(t_{i-1})}{t_i - t_{i-1}}$$

- It can be computed from position as:

$$\boldsymbol{a}^k(t_{i-1}) = \frac{\boldsymbol{p}^k(t_i) - 2\boldsymbol{p}^k(t_{i-1}) + \boldsymbol{p}^k(t_{i-2})}{(t_i - t_{i-1})^2}$$

- Magnitude of acceleration is also relevant:

$$a^k(t_i) = \sqrt{\left(a_x^k(t_i)\right)^2 + \left(a_y^k(t_i)\right)^2 + \left(a_z^k(t_i)\right)^2}$$

79

Group 1: kinematics

casa Paganini infomus

- Jerk is the rate of change of acceleration:

$$\boldsymbol{j}^k(t_i) = \frac{\boldsymbol{a}^k(t_i) - \boldsymbol{a}^k(t_{i-1})}{t_i - t_{i-1}}$$

- It can be computed from position as:

$$\boldsymbol{j}^k(t_{i-2}) = \frac{\boldsymbol{p}^k(t_i) - 2\boldsymbol{p}^k(t_{i-1}) + 2\boldsymbol{p}^k(t_{i-3}) + \boldsymbol{p}^k(t_{i-4})}{2(t_i - t_{i-1})^3}$$

- Magnitude of jerk describes movement (non)smoothness:

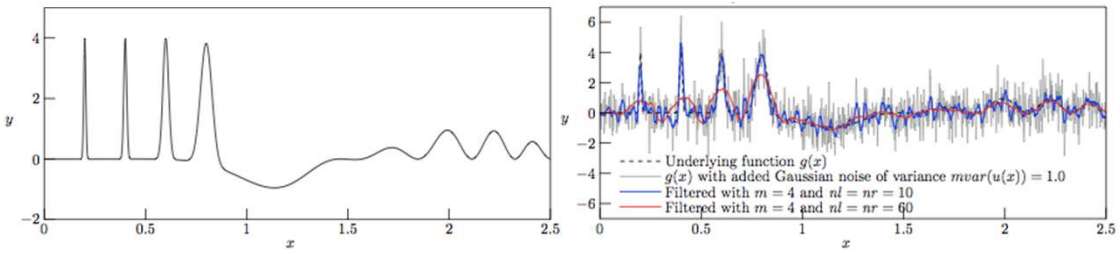
$$j^k(t_i) = \sqrt{\left(j_x^k(t_i)\right)^2 + \left(j_y^k(t_i)\right)^2 + \left(j_z^k(t_i)\right)^2}$$

80

Group 1: kinematics

casa Paganini infomus

- Differentiation degrades signal-to-noise ratio, unless the differentiation algorithm includes smoothing, carefully optimized for each application!
- The Savitzky-Golay filter combines differentiation and smoothing into one single algorithm.



81

Group 2: shape of trajectories

casa Paganini infomus

- Direction of movement is the unit vector of velocity:

$$\boldsymbol{d}^k(t_i) = \frac{\boldsymbol{p}^k(t_i) - \boldsymbol{p}^k(t_{i-1})}{\|\boldsymbol{p}^k(t_i) - \boldsymbol{p}^k(t_{i-1})\|}$$

- Curvature measures deviations from a straight line:

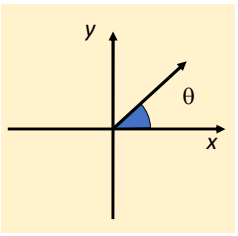
$$c^k(t_i) = \frac{\|\boldsymbol{a}^k(t_i) \times \boldsymbol{v}^k(t_i)\|}{\left(v^k(t_i)\right)^3}$$

- The radius of curvature is computed as $r^k = 1 \,/\, c^k(t_i)$.

82

Movement in specific directions

- Let us consider directions at time t_i: $\mathrm{D}(t_i) = \{\boldsymbol{d}^1(t_i), \boldsymbol{d}^2(t_i), \dots, \boldsymbol{d}^{M_p}(t_i)\}$.
- In 2D, for example, being $\theta(t_i)$ the angle between the *x* axis and the average direction unity vector, i.e., $\mathrm{avg}(\mathrm{D}(t_i))$, we detect:
  - Upward movement if $0° < \theta(t_i) < 180°$
  - Downward movement if $180° < \theta(t_i) < 360°$
  - Rightward movement if $-90° < \theta(t_i) < 90°$
  - Leftward movement if $90° < \theta(t_i) < 270°$
- Positive emotions may display upward movement (Boone and Cunningham, 1998).



83

Group 3: amount of movement

- Quantity of Motion (QoM, Larboulette and Gibet, 2014) is computed as the weighted sum of the speeds of a set of $\mathrm{K} \leq M_p$ landmarks $\mathrm{L}(t_i) \subseteq \{\boldsymbol{p}^1(t_i), \boldsymbol{p}^2(t_i), \dots, \boldsymbol{p}^{M_p}(t_i)\}$:

$$qom^{\mathrm{L}}(t_i) = \frac{\sum_{\boldsymbol{p}^j \in \mathrm{L}(t_i)} w_j \, v^j(t_i)}{\sum_{j=1}^{K} w_j}$$

- Typical subsets of landmarks include those corresponding to the arm region, head region, upper body, and lower body.

84

Kinetic energy

casa Paganini infomus

- Kinetic energy is computed as:

$$ke(t_i) = \frac{1}{2}\sum_{j=1}^{M_p} m_j \left(v^j(t_i)\right)^2$$

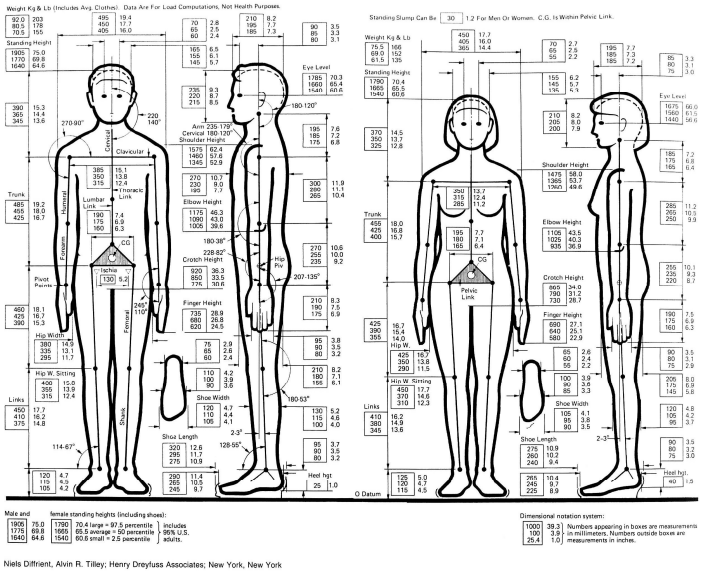
where m_j is the portion of mass associated with landmark $\boldsymbol{p}^j$ and $v^j(t_i)$ is its speed at time t_i.

- Portions of mass can be approximated by using data available in anthropometric tables.

85

Anthropometric tables: sizes

casa Paganini infomus



86

Anthropometric tables: weights

	Michigan data (Dempster)		Leipzig data (Braune and Fischer)	
	Mean weights (gm)	% of Total	Mean weights (gm)	% of Total
Total body weight	61190 ±8137 (7)	100	58895 (5)	
Head and trunk	34637 ±5607 (18)	56.34 ±2.45 (6)	30824 (5)	52.21 ±2.97(5)
Head and trunk minus shoulders	28077 ±3994 (18)	46.02 ±2.239(5)		
Head and neck	5119 ± 838 (16)	7.92 ±0.85 (7)		
Shoulders	3401 ± 843 (34)	5.27 ±0.546(14)		
Thorax	7669 ±2270 (17)	10.97 ±1.521(7)		
Abdomino-pelvic headless trunk	16318 ±2505 (18)	26.39 ±2.908(7)		
Arm	1636 ± 350 (42)	2.64 ±0.294(14)	2017 ±406 (22)	3.167±0.27 (10)
Forearm	947 ± 199 (42)	1.531±0.166(14)	1342 ±242 (18)	2.087±0.245(6)
Hand	378.3± 71.7(42)	0.612±0.058(14)	536.1± 84.4(18)	0.833±0.045(6)
Thigh	609.8± 985 (41)	10.008±1.197(14)	6632 ±783 (22)	10.924±0.769(10)
Shank	2852 ± 695 (41)	4.612±0.534(14)	2924 ±379 (22)	4.680±0.353(10)
Foot	884 ± 178 (41)	1.431±0.142(14)	1072 ±106 (22)	1.765±0.194(10)

Dempster, W. T., and Gaughran, G. R. L., 1967. Properties of body segments based on size and weight. American Journal of Anatomy, 120, 1, 33-54.

87

Group 4: use of space

- This is represented as the size over time of the bounding volume (BV) of a set of body landmarks.
- Given the set $\mathrm{L}(t_i) \subseteq \{\boldsymbol{p}^1(t_i), \boldsymbol{p}^2(t_i) \dots, \boldsymbol{p}^{M_p}(t_i)\}$ of $\mathrm{K} \leq M_p$ body landmarks, the size of its bounding volume $bv^{\mathrm{L}}(t_i)$ is:

$$bv^{\mathrm{L}}(t_i) = d_x^{\mathrm{L}}(t_i) \cdot d_y^{\mathrm{L}}(t_i) \cdot d_z^{\mathrm{L}}(t_i)$$

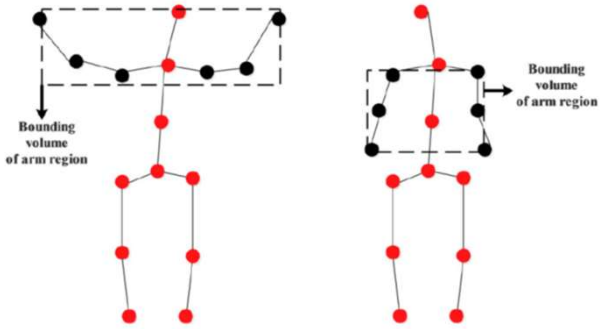
where:

- $d_x^{\mathrm{L}}(t_i) = \left| \max_{\boldsymbol{p}^{\mathrm{j}} \in \mathrm{L}(t_i)} x^j(t_i) - \min_{\boldsymbol{p}^{\mathrm{j}} \in \mathrm{L}(t_i)} x^j(t_i) \right|$
- $d_y^{\mathrm{L}}(t_i) = \left| \max_{\boldsymbol{p}^{\mathrm{j}} \in \mathrm{L}(t_i)} y^j(t_i) - \min_{\boldsymbol{p}^{\mathrm{j}} \in \mathrm{L}(t_i)} y^j(t_i) \right|$
- $d_z^{\mathrm{L}}(t_i) = \left| \max_{\boldsymbol{p}^{\mathrm{j}} \in \mathrm{L}(t_i)} z^j(t_i) - \min_{\boldsymbol{p}^{\mathrm{j}} \in \mathrm{L}(t_i)} z^j(t_i) \right|$

88

Group 4: use of space

- BV is the volume of the smallest parallelepiped containing the selected body landmarks.
- It is typically computed for the arm region, the head region, the upper body, the lower body, and the whole body.



Source: Ahmed et al., 2020.

89

Group 5: displacements

- These are computed as the distances of body landmarks corresponding to major joints from a reference landmark.
- For each relevant landmark $\boldsymbol{p}^h(t_i)$, displacement from the reference landmark $\boldsymbol{p}^r(t_i)$ is computed as:

$$ds^h(t_i) = \left\|\boldsymbol{p}^h(t_i) - \boldsymbol{p}^r(t_i)\right\|$$

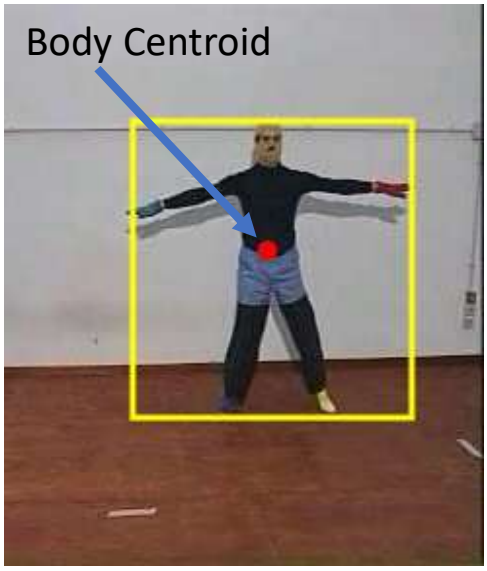
- In (Ahmed, 2020), the considered landmarks $\boldsymbol{p}^h$ are those corresponding to head, neck, shoulder, elbow, wrist, hand, knee, ankle, foot, and the body centroid.
- The base of the spine is taken as reference landmark $\boldsymbol{p}^r$.

90

Body centroid

casa Paganini infomus

- The body centroid or Center of Gravity is considered as the central point of a pose.
- This is obtained by computing the 1st order moments over the coordinates of the body landmarks representing the pose.
- If landmarks represent mass, then the 1st order moment divided by the total mass (i.e., the 0th order moment) is the Center of Mass.



91

Body centroid

casa Paganini infomus

- The body centroid of a pose is thus computed as:

$$\boldsymbol{p}^c(t_i) = \left(x^c(t_i), y^c(t_i), z^c(t_i)\right) \qquad \text{where:}$$

$$x^c(t_i) = \frac{1}{M_p}\sum_{j=1}^{M_p} x^j(t_i) \qquad y^c(t_i) = \frac{1}{M_p}\sum_{j=1}^{M_p} y^j(t_i)$$

$$z^c(t_i) = \frac{1}{M_p}\sum_{j=1}^{M_p} z^j(t_i)$$

92

Group 6: compactness

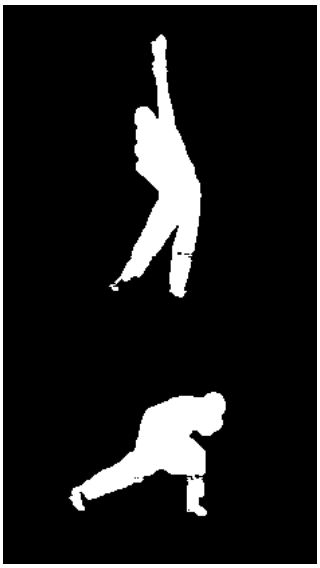
casa Paganini infomus

- This is a collection of features describing the extent to which limbs are extended or kept close to the body centroid.
- Verticality. Maximum value of the *y* component:

$$vt(t_i) = \max_{j=1\ldots M_p} y^j(t_i)$$

- Extension. Maximum distance from the body centroid:

$$ex(t_i) = \max_{j=1\ldots M_p} \left\| \boldsymbol{p}^j(t_i) - \boldsymbol{p}^c(t_i) \right\|$$



Two poses with different verticality

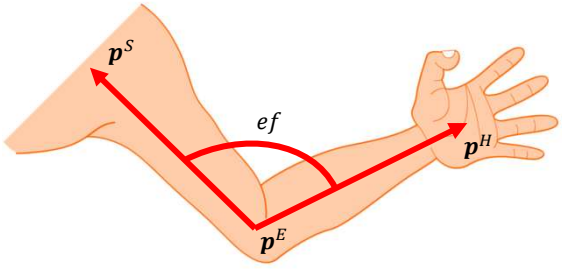
93

Group 6: compactness

casa Paganini infomus

- Elbow flexion. Angle formed by shoulder (landmark $\boldsymbol{p}^S$), elbow (landmark $\boldsymbol{p}^E$), and hand (landmark $\boldsymbol{p}^H$) :

$$ef(t_i) = \cos^{-1}\left(\frac{\left(\boldsymbol{p}^S(t_i) - \boldsymbol{p}^E(t_i)\right)\cdot\left(\boldsymbol{p}^H(t_i) - \boldsymbol{p}^E(t_i)\right)}{\left\|\boldsymbol{p}^S(t_i) - \boldsymbol{p}^E(t_i)\right\|\left\|\boldsymbol{p}^H(t_i) - \boldsymbol{p}^E(t_i)\right\|}\right)$$



94

Group 6: compactness

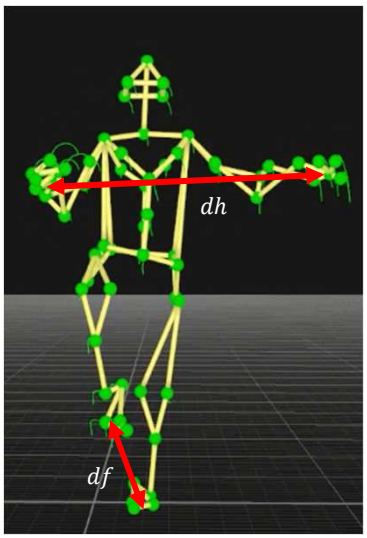
casa Paganini infomus

- Arm shape. Magnitude of the vector from hand (taken as landmark $\boldsymbol{p}^H$) to base of the spine (landmark $\boldsymbol{p}^{SP}$) :

$$as(t_i) = \|\boldsymbol{p}^H(t_i) - \boldsymbol{p}^{SP}(t_i)\|$$

- Hands and feet relationships. Distances between the two hands (landmarks $\boldsymbol{p}^{RH}$ and $\boldsymbol{p}^{LH}$) and between the two feet (landmarks $\boldsymbol{p}^{RF}$ and $\boldsymbol{p}^{LF}$) :

$$dh(t_i) = \|\boldsymbol{p}^{RH}(t_i) - \boldsymbol{p}^{LH}(t_i)\|$$
$$df(t_i) = \|\boldsymbol{p}^{RF}(t_i) - \boldsymbol{p}^{LF}(t_i)\|$$



Distances between hands and feet

95

Other features for compactness

casa Paganini infomus

- Point density is computed as the average distance between the body landmarks and the body centroid:

$$pd(t_i) = \frac{1}{M_p}\sum_{j=1}^{M_p}\left\|\boldsymbol{p}^j(t_i) - \boldsymbol{p}^c(t_i)\right\|$$

- A 3D contraction index is computed as the ratio between the bounding volume of the torso bv^{T} and the bounding volume of the whole body bv^{P} (Piana et al. 2016), i.e.:

$$ci(t_i) = \frac{bv^{\mathrm{T}}(t_i)}{bv^{\mathrm{P}}(t_i)}$$

96

Other features for compactness

casa Paganini infomus

- A contraction index of a pose $\mathrm{P}(t_i)$ can also be computed by:
  - Retrieving the minimum-volume ellipsoid $\mathcal{E}^{\mathrm{L}}(t_i)$ enclosing pose $\mathrm{P}(t_i)$ or a subset thereof $\mathrm{L}(t_i) \subseteq \mathrm{P}(t_i)$.
  - Calculating its sphericity, i.e., how close $\mathcal{E}^{\mathrm{L}}(t_i)$ is to a sphere.
- The minimum-volume ellipsoid $\mathcal{E}^{\mathrm{L}}(t_i)$ can be obtained by applying the Khachiyan algorithm (Khachiyan, 1996).
- Sphericity is computed as:

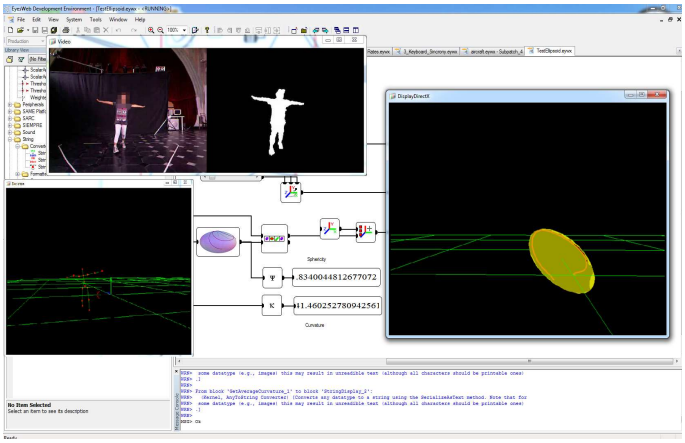
$$\Psi^{\mathrm{L}}(t_i) = \frac{\pi^{\frac{1}{3}}\left(6V_{\mathcal{E}^{\mathrm{L}}(t_i)}\right)^{\frac{2}{3}}}{A_{\mathcal{E}^{\mathrm{L}}(t_i)}}$$

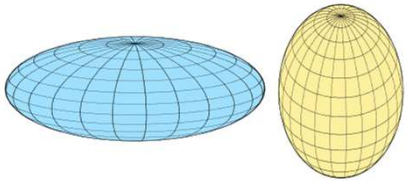
where V is the volume and A the surface area of $\mathcal{E}^{\mathrm{L}}(t_i)$.

97

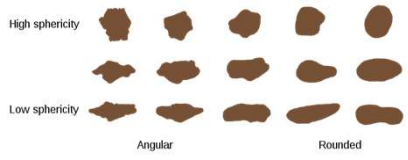
Other features for compactness

casa Paganini infomus





Source: Wikipedia. Author: Tomruen



Source: Wikipedia.

- Example: computation of the minimum-volume ellipsoid and of its sphericity by means of the EyesWeb XMI platform.

98

Group 7: Laban's Effort

casa Paganini infomus

- In his Theory of Effort, choreographer Rudolf Laban points out the dynamic nature of movement and the relationship among movement, space, and time.
- Laban's approach is an attempt to describe, in a formalized way, the major features of human movement without focusing on a particular kind of movement or expression.
- Extraction and analysis of Laban's gesture qualities is a step toward analysis and understanding of expressive gesture.

99

Group 7: Laban's Space

casa Paganini infomus

- Space refers to the direction of a movement stroke and to the path followed by a sequence of strokes. If movement follows these directions smoothly, space is flexible; whilst if it follows them along a straight trajectory, space is direct.

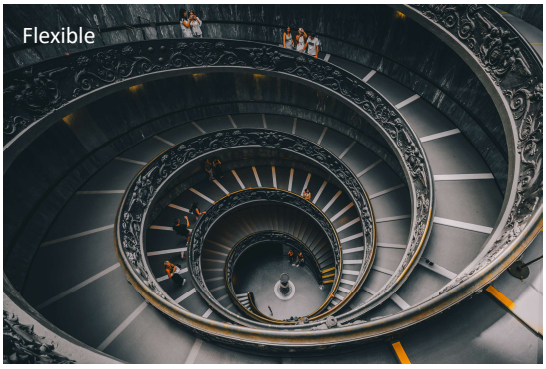


Photo by Grillot edouard on Unsplash

Photo by Daniel Eledut on Unsplash

100

Group 7: Laban's Space

casa Paganini infomus

- For body landmark $\boldsymbol{p}^k$, this can be computed as the ratio between the distance between the first and last point in the trajectory of the landmark and its length over a temporal window (also called Directness Index):

$$ls^k(t_i) = \frac{\left\|\boldsymbol{p}^k(t_i) - \boldsymbol{p}^k(t_{i-\mathrm{W}})\right\|}{\sum_{j=0}^{W-1}\left\|\boldsymbol{p}^k(t_{i-j}) - \boldsymbol{p}^k(t_{i-j-1})\right\|}$$

  where W is the number of previous poses that are taken into account in the computation. It must be $i - W \geq 0$.
- So, the values of $ls^k(t_i)$ will be available on and after t_W.

101

Group 7: Laban's Time

casa Paganini infomus

- Time is related to "urgency", to impulsiveness and to the capacity of controlling a movement.
- With respect to Time an action can be quick or sustained.

Photo by Edoardo Busti on Unsplash

Photo by Thomas Tucker on Unsplash

102

Group 7: Laban's Time

casa Paganini infomus

- Larboulette and Gibet (2014) propose to compute Laban's Time for body landmark $\boldsymbol{p}^k$ as the average of the magnitude of its acceleration over a temporal window, that is:

$$lt^k(t_i) = \frac{1}{W}\sum_{j=1}^{W} a^k\left(t_{i-j+1}\right)$$

  where W is the number of previous poses that are taken into account in the computation. It must be $i - W \geq 0$.
- Again, the values of $lt^k(t_i)$ will be available on and after t_W.

Larboulette, C., and Gibet, S. 2015. A review of computable expressive descriptors of human motion. In Proceedings of the 2nd International Workshop on Movement and Computing (MOCO'15), 21–28.

103

Group 7: Laban's Weight

casa Paganini infomus

- Weight is a measure of how much strength and weight is exerted in a movement. It can be strong or light. For example, in pushing away a heavy object a strong weight is used, whereas in handling a delicate object, weight is light.

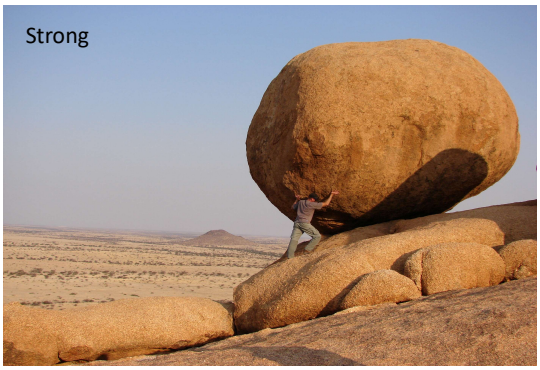


Photo by Vicky Sim on Unsplash

Photo by Jeffrey Hamilton on Unsplash

104

Group 7: Laban's Weight

casa Paganini infomus

- Larboulette and Gibet (2014) propose to compute overall Laban's Weight as the maximum of kinetic energy over a temporal window, that is:

$$lw(t_i) = \max_{j=i-W+1 \ldots i} ke(t_j)$$

where W is the number of previous poses that are taken into account in the computation. It must be $i - W \geq 0$.

- The values of $lw(t_i)$ will be available on and after t_W.

105

Group 7: Laban's Flow

casa Paganini infomus

- Flow is a measure of how bound or free a movement, or a sequence of movements, appears.
- Flow was not included in the initial version of Laban's Theory of Effort and was added later on, at a second stage.

Photo by Thao Le Hoang on Unsplash

Photo by Persnickety Prints on Unsplash

106

Group 7: Laban's Flow

casa Paganini infomus

- Larboulette and Gibet (2014) propose to compute Laban's Flow for body landmark $\boldsymbol{p}^k$ as the average of the magnitude of jerk over a temporal window, that is:

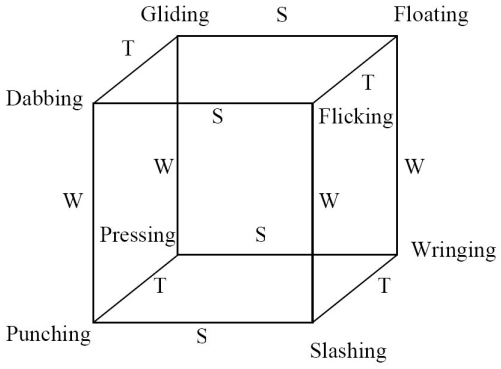
$$lf^k(t_i) = \frac{1}{W}\sum_{j=1}^{W} j^k\left(t_{i-j+1}\right)$$

  where W is the number of previous poses that are taken into account in the computation. It must be $i - W \geq 0$.
- The values of $lf^k(t_i)$ will be available on and after t_W.

107

Laban's Theory of Effort

casa Paganini infomus



Basic Effort	Space	Time	Weight
Pressing	*Direct*	*Sustained*	*Strong*
Flicking	*Flexible*	*Sudden*	*Light*
Punching	*Direct*	*Sudden*	*Strong*
Floating	*Flexible*	*Sustained*	*Light*
Wringing	*Flexible*	*Sustained*	*Strong*
Dabbing	*Direct*	*Sudden*	*Light*
Slashing	*Flexible*	*Sudden*	*Strong*
Gliding	*Direct*	*Sustained*	*Light*

108

Group 8: bounding triangle

casa Paganini infomus

- This group of features concerns the bounding triangle (Glowinski et al., 2011), i.e., the triangle formed by the two hands (landmarks $\boldsymbol{p}^{RH}$ and $\boldsymbol{p}^{LH}$) and the head (landmark $\boldsymbol{p}^{H}$).
- Its barycenter is computed as:

$$\boldsymbol{p}^{tc}(t_i) = \frac{1}{3}\left(\boldsymbol{p}^{RH}(t_i) + \boldsymbol{p}^{LH}(t_i) + \boldsymbol{p}^{H}(t_i)\right)$$

- Its spatial extent is computed as:

$$te(t_i) = \|\boldsymbol{p}^{tc}(t_i) - \boldsymbol{p}^{r}(t_i)\|$$

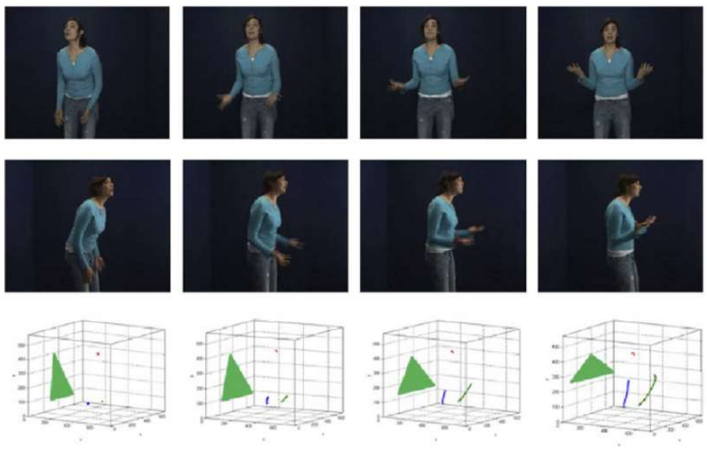
- The base of the spine was taken as reference landmark $\boldsymbol{p}^{r}$.

109

Group 8: bounding triangle

casa Paganini infomus

- Example: the bounding triangle extracted from a video excerpt of the GEMEP corpus (Glowinski et al, 2011).



Source: Glowinski, D., Dael, N., Camurri, A., Volpe, G., Mortillaro, M., and Scherer, K., 2011. Toward a Minimal Representation of Affective Gestures. IEEE Transactions on Affective Computing, 2(2), 106-118.

110

Group 9: symmetry

casa Paganini infomus

- Two ways for computing symmetry (Glowinski et al, 2011).
- The first one refers to symmetry of the bounding triangle (hands $\boldsymbol{p}^{RH}$ and $\boldsymbol{p}^{LH}$, and barycenter $\boldsymbol{p}^{tc}$). A horizontal and a vertical symmetry index are computed and then aggregated:

$$hs(t_i) = \frac{\left||x^{tc}(t_i) - x^{LH}(t_i)| - |x^{tc}(t_i) - x^{RH}(t_i)|\right|}{|x^{tc}(t_i) - x^{LH}(t_i)| + |x^{tc}(t_i) - x^{RH}(t_i)|}$$

$$vs(t_i) = \frac{\left||y^{tc}(t_i) - y^{LH}(t_i)| - |y^{tc}(t_i) - y^{RH}(t_i)|\right|}{|y^{tc}(t_i) - y^{LH}(t_i)| + |y^{tc}(t_i) - y^{RH}(t_i)|}$$

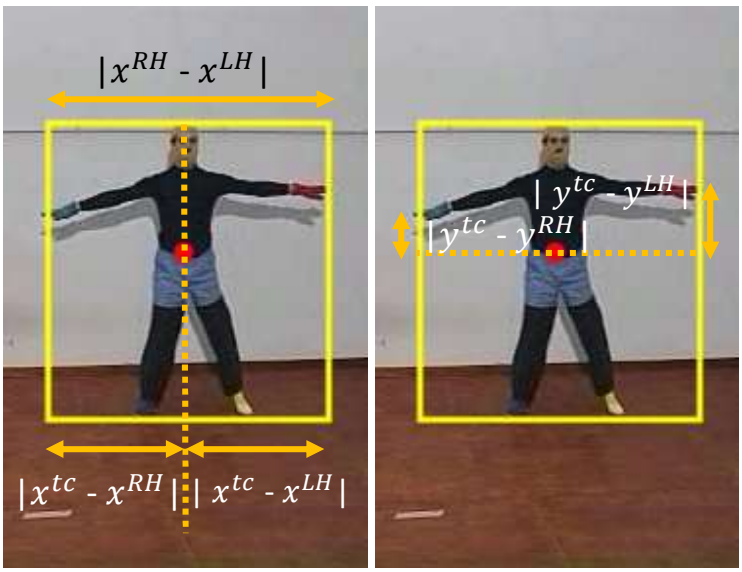
$$si_1(t_i) = \frac{hs(t_i)}{vs(t_i)}$$

111

Group 9: symmetry

casa Paganini infomus

Horizontal and vertical symmetry indexes.



112

Group 9: symmetry

casa Paganini infomus

- The second one grounds on the concept of geometric entropy.
- This is a measure of the extent to which a trajectory is spread over space (e.g., Cordier et al., 1994). For body landmark $\boldsymbol{p}^k$:

$$h^k(t_i) = \ln \frac{2tl^k(t_i)}{pc^k(t_i)}$$

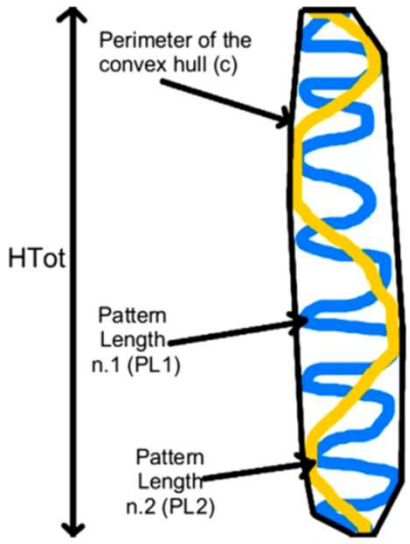
where:

- $tl^k(t_i)$ is the length of the trajectory of $\boldsymbol{p}^k$ over a temporal window W, i.e.: $tl^k(t_i) = \sum_{j=0}^{W-1} \left\| \boldsymbol{p}^k(t_{i-j}) - \boldsymbol{p}^k(t_{i-j-1}) \right\|$.
- $pc^k(t_i)$ is the length of the perimeter of the convex hull of the trajectory (i.e., of the smallest polygon enclosing the trajectory).

113

Group 9: symmetry

casa Paganini infomus



Geometric Entropy (H):

H1(blue) = ln(2*PL1/c)

H2(orange) = ln(2*PL2/c)

since PL1 > PL2
H1 > H2
for the same height HTot

A graphic explanation of geometric entropy.

Source: Sibella, F., Frosio, I., Schena, F., Borghese, N.A., 2007. 3D analysis of the body center of mass in rock climbing. Human Movement Science, 26(6), 841-852.

114

Group 9: symmetry

- The projection on the frontal plane of the trajectories of the landmarks $\boldsymbol{p}^{LH}$ and $\boldsymbol{p}^{RH}$ of the two hands is computed.
- The entropies $h^{LH}(t_i)$ and $h^{RH}(t_i)$ of the projected 2D trajectories are calculated over a temporal window W.
- Symmetry is finally obtained as the ratio of such entropies:

$$si_2(t_i) = \frac{h^{LH}(t_i)}{h^{RH}(t_i)}$$

- The two indexes represent different facets of symmetry, the first being a measure of postural symmetry and the second one considering temporal development of movement.

115

Group 10: balance

- This group of features describes how balance of various segments of the human body changes during movement.
- Ahmed (2020) suggests computing (i) displacement of the body centroid and (ii) balance as the difference of the positions of the centroids of the upper body ($\boldsymbol{p}^{uc}(t_i)$) and of the lower body ($\boldsymbol{p}^{lc}(t_i)$):

$$ds^c(t_i) = \|\boldsymbol{p}^c(t_i) - \boldsymbol{p}^c(t_{i-1})\|$$

$$bl_1(t_i) = \|\boldsymbol{p}^{uc}(t_i)\| - \|\boldsymbol{p}^{lc}(t_i)\|$$

116

Group 10: balance

casa Paganini infomus

- Larboulette and Gibet (2014) compute balance as a binary value indicating whether the projection of the body centroid on the ground lies within the surface of the support polygon.
- This is the region the center of mass must lie over for stability. For an object resting on a horizontal surface, it is computed as the convex hull of the contact points of the object with the surface. Balance is thus obtained as:

$$bl_2(t_i) = \mathbf{1}_{sp(t_i)}(\mathrm{P} \cdot \boldsymbol{p}^c(t_i))$$

where P is the applied projection matrix, sp is the support polygon, and $\mathbf{1}_{sp(t_i)}: \mathbb{R}^2 \longmapsto \{0, 1\}$ is the indicator function of $sp(t_i) \subset \mathbb{R}^2$.

117

Output time-series

casa Paganini infomus

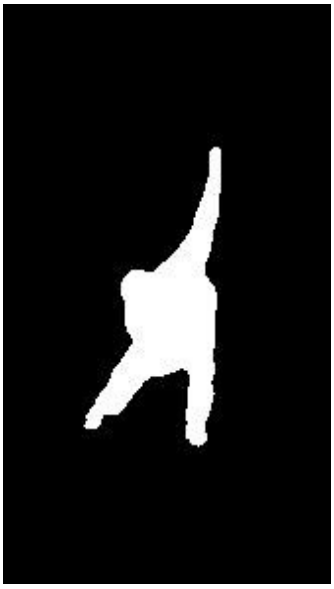
- The output time-series $\mathrm{F} = \{\mathrm{F}_1, \mathrm{F}_2, \dots, \mathrm{F}_\mathrm{T}\}$ are built by taking the values of the features at each time $t_i, i = 1, \dots, N$, e.g.:
  - Velocity of landmark $\boldsymbol{p}^k$: $\boldsymbol{V}^k = (\boldsymbol{v}^k(t_2), \boldsymbol{v}^k(t_3), \dots, \boldsymbol{v}^k(t_N))$
  - Speed of landmark $\boldsymbol{p}^k$: $S^k = (v^k(t_2), v^k(t_3), \dots, v^k(t_N))$
  - Acceleration of landmark $\boldsymbol{p}^k$: $\boldsymbol{A}^k = (\boldsymbol{a}^k(t_3), \boldsymbol{a}^k(t_4), \dots, \boldsymbol{a}^k(t_N))$
  - Jerk of landmark $\boldsymbol{p}^k$: $\boldsymbol{J}^k = (\boldsymbol{j}^k(t_4), \boldsymbol{j}^k(t_5), \dots, \boldsymbol{j}^k(t_N))$
  - Direction of landmark $\boldsymbol{p}^k$: $\boldsymbol{D}^k(t_i) = (\boldsymbol{d}^k(t_2), \boldsymbol{d}^k(t_3), \dots, \boldsymbol{d}^k(t_N))$
  - Curvature of landmark $\boldsymbol{p}^k$: $C^k(t_i) = (c^k(t_3), c^k(t_4), \dots, c^k(t_N))$
  - …

118

Blob features

casa Paganini infomus

- The 10 groups of features are computed on 3D trajectories of body landmarks.
- Most of them can also be calculated on 2D trajectories, if these are the only available data.
- What about if we just have blobs?



119

Blob features

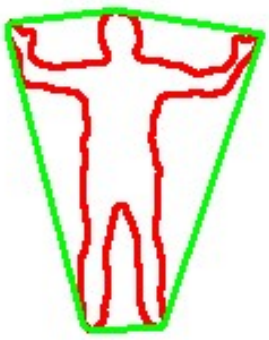
casa Paganini infomus

- One possibility is to extract the coordinates of some relevant bidimensional points from the blob.

Body centroid

Vertexes of the bounding rectangle

Vertexes of the convex hull



Centroids of parts of the body

120

Blob features

casa Paganini infomus

- Another possibility is to process the blob to calculate a set of blob features.
- These cover some of the groups of features. A few examples:
  - Group 3, amount of motion: Motion Index.
  - Group 4, use of space: Body Shape
  - Group 6, compactness: Contraction Index
  - Group 9, symmetry: Asymmetry Index

Camurri, A., Mazzarino, B., and Volpe, G., 2004. Analysis of Expressive Gesture: The EyesWeb Expressive Gesture Processing Library. In Camurri, A., and Volpe, G. (Eds.), Gesture-based Communication in Human-Computer Interaction, LNAI 2915, Springer Verlag, 2004, pp. 460-467

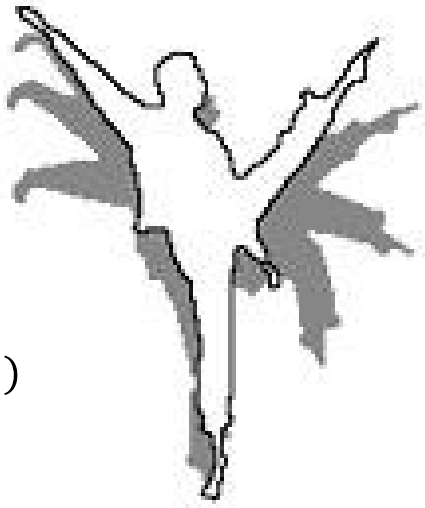
121

Silhouette Motion Images

casa Paganini infomus

- SMIs (Silhouette Motion Images) carry information on variations of a blob (usually the silhouette of a user) in the last *W* frames.

$$SMI(t_i) = \left(\sum_{k=1}^{W} \mathrm{B}(t_{i-k}) \right) - \mathrm{B}(t_i)$$



122

Motion Index

casa Paganini infomus

- SMIs carry information about the detected movement.
- SMI area can be taken as a measure of amount of motion.
- The SMI area, normalized by the blob area, is called Motion Index (or Quantity of Motion).

$$mi(t_i) = \frac{Area(SMI(t_i))}{Area(\mathrm{B}(t_i))}$$

- Note that this is an approximated measure: e.g., movement against the video-camera is not detected.

123

Weighted Motion Index

casa Paganini infomus

- Motion Index can also be computed by weighting pixels differently in the input blob.
- So, we can compute a Motion Index where pixels close to the centre of the blob weight more than pixels close to the contour (i.e., something approaching kinetic energy).
- Or we can compute a Motion Index where pixels close to the contour weight more than pixels close to the centre of the blob (i.e., a kind of perceptual measure where limbs have a stronger impact on perception of movement).

124

Body Shape

casa Paganini infomus

- Being $\{(x^1, y^1), (x^2, y^2), \ldots, (x^{M_b}, y^{M_b})\}$ the 2D coordinates of the M_b pixels belonging to a blob, second order central moments are used to obtain an elliptical approximation of the shape of the blob. These are computed as:
  - $\mu_{2,0}(t_i) = \frac{1}{M_b}\sum_{j=1}^{M_b}\left(x^j(t_i) - x^c(t_i)\right)^2$
  - $\mu_{0,2}(t_i) = \frac{1}{M_b}\sum_{j=1}^{M_b}\left(y^j(t_i) - y^c(t_i)\right)^2$
  - $\mu_{1,1}(t_i) = \frac{1}{M_b}\sum_{j=1}^{M_b}\left(x^j(t_i) - x^c(t_i)\right)\left(y^j(t_i) - y^c(t_i)\right)$

  where (x^c, y^c) are the coordinates of the body (blob) centroid.

125

Body Shape

casa Paganini infomus

- The angle $\theta(t_i)$ between the major axis of the approximating ellipse and the *x* axis of the reference system is associated with body orientation (e.g., rightward or leftward leaning).
- Eccentricity $\varepsilon(t_i)$ is related to contraction/expansion.

$$\theta(t_i) = \frac{1}{2}\tan^{-1}\frac{2\mu_{1,1}(t_i)}{\mu_{2,0}(t_i) - \mu_{0,2}(t_i)}$$

$$\varepsilon(t_i) = \frac{\left(\mu_{2,0}(t_i) - \mu_{0,2}(t_i)\right)^2 - 4\mu_{1,1}^2(t_i)}{\left(\mu_{2,0}(t_i) + \mu_{0,2}(t_i)\right)^2}$$

126

Contraction Index

casa Paganini infomus

- Contraction Index is a measure of the extent to which a body pose is contacted, and is computed as the ratio between the area of the blob and the area of its bounding rectangle:

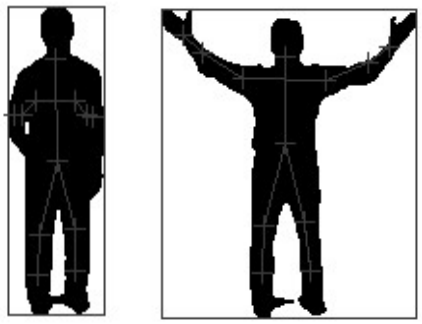
$$ci(t_i) = \frac{Area(\mathrm{B}(t_i))}{br^B(t_i)}$$

where:

$$br^B(t_i) = d_x^{\mathrm{B}}(t_i) \cdot d_y^{\mathrm{B}}(t_i)$$

$$d_x^{\mathrm{B}}(t_i) = \left| \max_{j=1\ldots M_b} x^j(t_i) - \min_{j=1\ldots M_b} x^j(t_i) \right|$$

$$d_y^{\mathrm{B}}(t_i) = \left| \max_{j=1\ldots M_b} y^j(t_i) - \min_{j=1\ldots M_b} y^j(t_i) \right|$$



127

Asymmetry Index

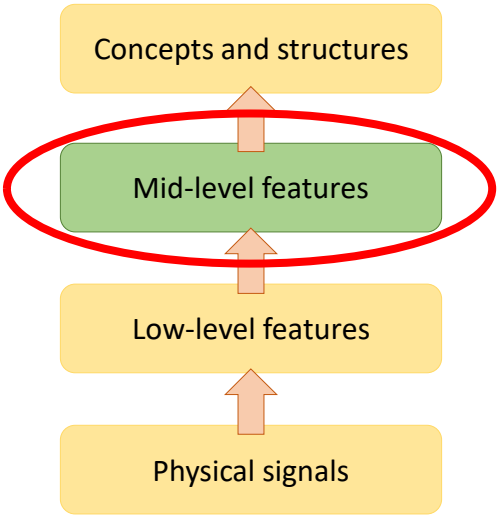
casa Paganini infomus

- Asymmetry Index measures asymmetry of a body pose with respect to the axes of its bounding rectangle.
- It is computed as follows:
  - Based on centroid divide blob in two halves.
  - Flip horizontally one half on the top of the other and compute the difference between the two halves.
  - Compute Asymmetry Index as the normalized area of the difference blob (the more the two halves are symmetric the more the difference blob will be small).

128

A conceptual framework

casa Paganini infomus

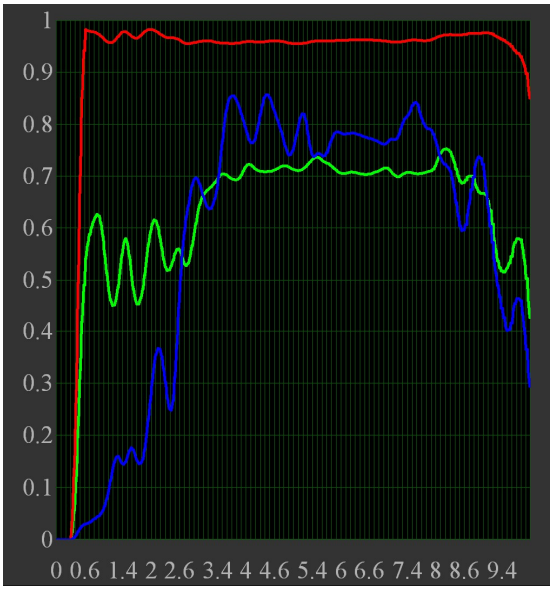


129

Input: time-series of sampled data

casa Paganini infomus

- Input of layer 3 is a set of T time-series $\mathrm{F} = \{f_1, f_2 \ldots f_\mathrm{T}\}$, each of them representing a feature characterizing movement over time.
- Data in the time-series is sampled at a rate f_s that depends on the input device. Usually all time-series are sampled with the same rate.

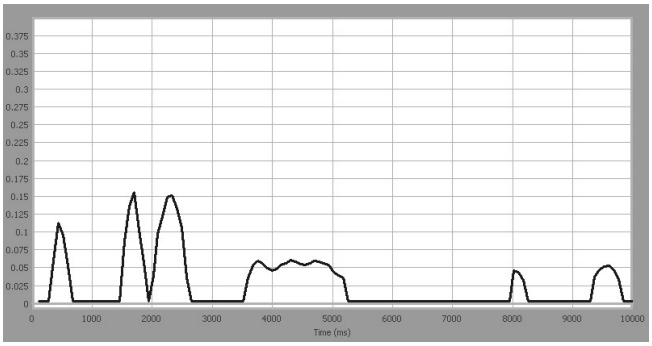


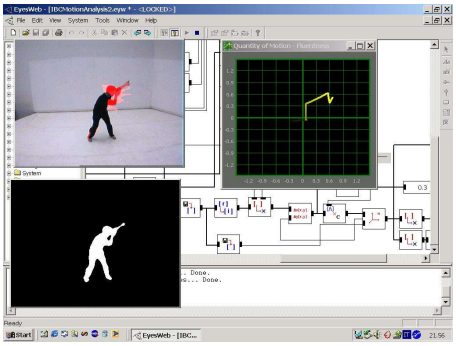
130

Processing

casa Paganini infomus

- Two basic steps:
  - Unitizing, i.e., identification of movement units.
  - Annotation, i.e., providing measures that characterize what is observed within each movement unit.





131

Unitizing

casa Paganini infomus

- Unitizing is a general process (i.e., it does not concern movement only) and consists of dividing an observation into discrete smaller units.
- It can be performed either manually or automatically.
- Unitizing is crucial:
  - For providing meaningful units to the annotation step.
  - For creating reliable ground truth material, i.e., a label can be given to each unit. Such labels can then be used when machine learning techniques are applied.

132

Unitizing: approaches

casa Paganini infomus

- Interval coding: units have a fixed-length interval of time.
  - Thin slices approach: fixed-length time windows of behavior from 2 seconds to 5 minutes are deemed to provide an efficient assessment of personality, affect, and interpersonal relations (e.g., as in Gatica-Perez et. al, 2015).
  - Interval coding is fast to perform and easy to automatize. It does not require any prior knowledge of the content of an interaction.
  - Nevertheless, it might lead to boundaries of the units being placed within actions, thus resulting in losing information.
  - It is not suitable for segmenting single gestures, but it can be applied when movement does not have clear pauses so that single units (gestures) cannot be identified easily.

133

Unitizing: approaches

casa Paganini infomus

- Continuous coding: every single utterance, gesture, and so on is identified, unitized, and annotated.
  - It look and sample for specific behaviors, either at scheduled or random points, throughout an interaction.
  - For example, ACT4Teams (Kauffeld et al., 2018) breaks an observation into thought units, i.e., the smallest meaningful segments of behavior that can be coded into 43 different categories. ACT4Teams is used for unitizing group behavior.
  - This approach can be tailored for specific objectives and studies.
  - Nevertheless, it is very difficult to automatize and very time consuming to perform manually.

134

Unitizing: approaches

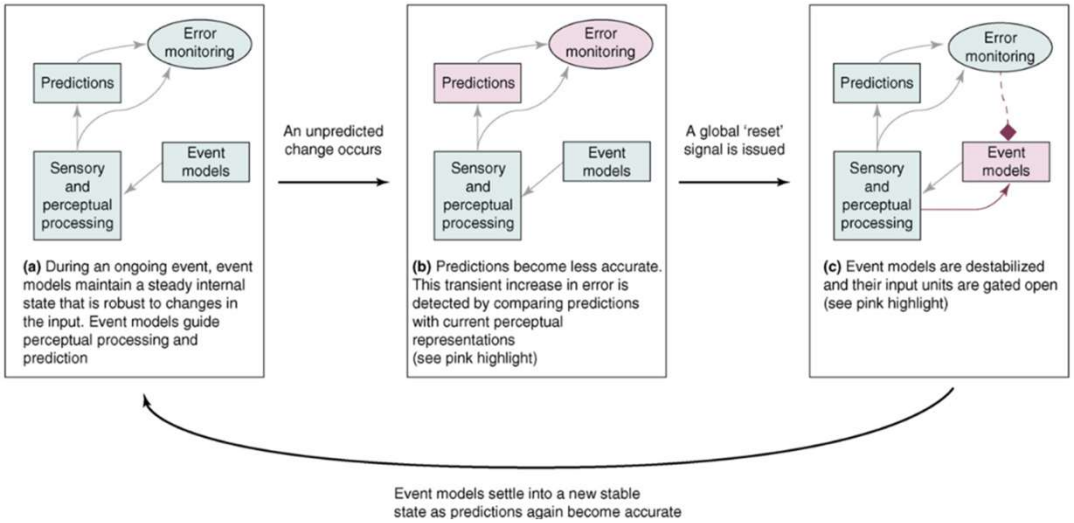
casa Paganini infomus

- Cognitive inspired segmentation: it grounds on the Event Segmentation Theory (Zacks, 2007):
  - An event is defined as "a segment of time at a given location, that is conceived by an observer to have a beginning and an end".
  - Segmentation relies on event models that observers form of the ongoing situation: a boundary is perceived whenever unpredictable changes occur, putting the current event model at stake.
  - Changes concern seven dimensions: time, space, objects, characters, character interaction, causes, goals.
  - This approach seems a viable alternative with respect to interval coding and continuous coding (Ceccaldi et al., 2019).
  - Open challenge: developing an automatic version of it!

135

Cognitive inspired segmentation

casa Paganini infomus



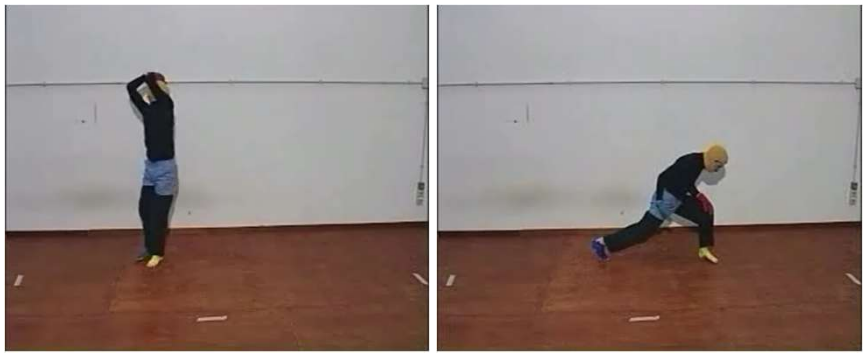
Source: Kurby, C. A., and Zacks, J. M., 2008. Segmentation in the perception and memory of events. Trends in cognitive sciences, 12(2), 72–79.

136

Unitizing applied to movement

- This consists of distinguishing between:
  - Movement units: the user is moving.
  - Pauses: the user does not appear to move.
- It is also referred to as movement segmentation.

A movement unit and a pause



137

Movement units and gestures

- A movement unit may correspond to a gesture.

138

Gesture

casa Paganini infomus

- "A natural gesture means the types of gestures spontaneously generated by a person telling a story, speaking in public, or holding a conversation" (Cassel et al., 1990).
- "Movements of the arms and hands which are closely synchronized with the flow of speech" (McNeill, 1992).
- "A gesture is a movement of one's body that conveys meaning to oneself or to a partner in communication" (Hummels et al., 1998);
- "A movement of the body that contains information" (Kurtenbach and Hulteen, 1990).

139

Gesture: functions

casa Paganini infomus

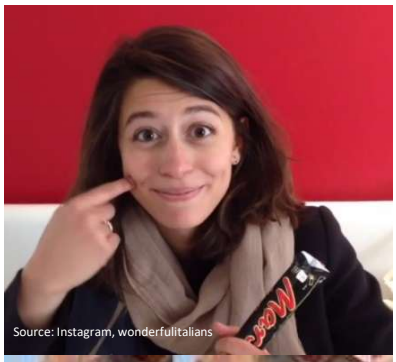
- McNeill's taxonomy (1992) distinguishes between:
  - Iconic gestures: air pictures representing aspects of the object being discussed, e.g., its shape.
  - Metaphoric gestures: represent abstract concepts or abstract features of an object. These gestures are diverse and vary with language or culture.
  - Deictic gestures are pointing motions, i.e., they identify the location of people, places, and things.
  - Beats are little waves of the hand that underscore the value of speech, give accent to words, and help in speaker turn-taking.
- Limitation: it focuses on natural gestures accompany speech.

140

McNeill's Taxonomy

Iconic gestures

Deictic gestures



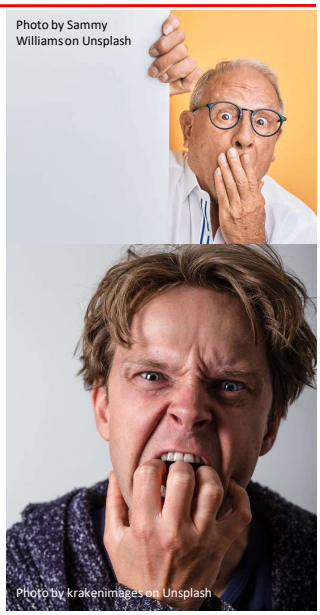
Metaphoric gestures

Beat gestures

141

Expressive gesture

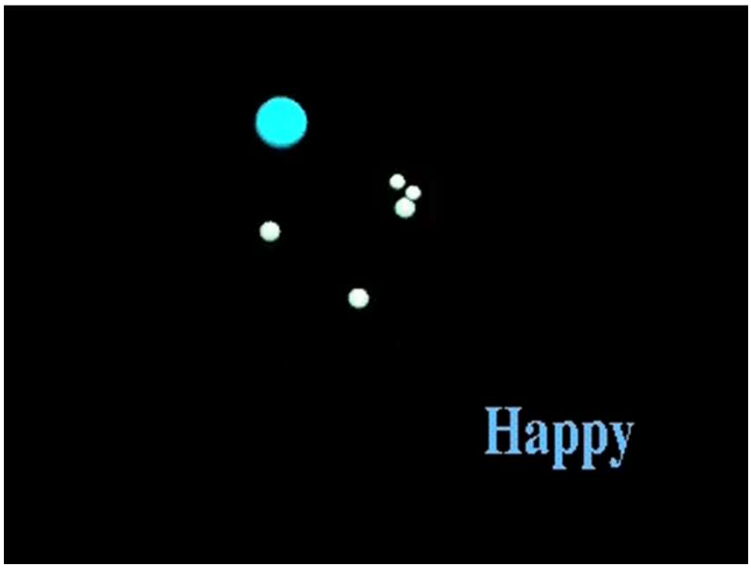
- Besides their role in denoting things and accompanying speech, gestures can also communicate emotions and support social interaction.
- Expressive gesture is conceived as a movement of the body that contains and conveys expressive information, i.e., information concerning emotion.
- This information is called expressive content and is independent from, even if superimposed to, a possible denotative meaning of the gesture.



142

Expressive content

casa Paganini infomus



Everyday actions (Pollick, 2001): door knocking

143

Expressive content

casa Paganini infomus

Two brief dance fragments (micro-dances) performed with different expressive qualities: fluid vs. rigid.

144

Unitizing applied to movement

casa Paganini infomus

- Computationally simple approaches to automatic unitizing of human movement include:
  - Thin slices: fixed-length time windows long enough to grab movement time evolution (e.g., 0.5 – 3s). Note that these time windows are much longer with respect to those used at layer 2.
  - Thresholding: units are identified by applying a threshold on energy, i.e., a boundary between units is detected when energy falls below the threshold.
- Thin slices are commonly applied when a movement stream does not have evident pauses so that movement units are hard to identify.

145

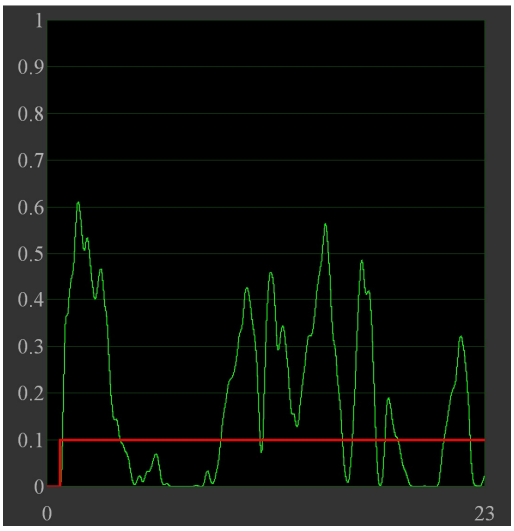
Unitizing by thresholding

casa Paganini infomus

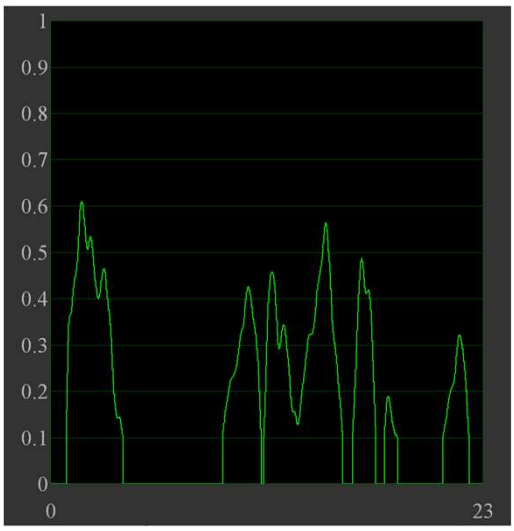
- It can be applied when a movement stream consists of a quite regular sequence of gestures and pauses.
- This approach is fast to perform and particularly suited for automatic unitizing.
- Either a fixed threshold or an adaptive one may be used.
- The approach was proved to reflect manual unitizing of the same movement stream in pause and movement phases, as performed by human participants (Glowinski et al., 2009).
- Nevertheless, it does not take into account cognitive and affective processes (e.g., formation and change of goals).

146

Unitizing by thresholding



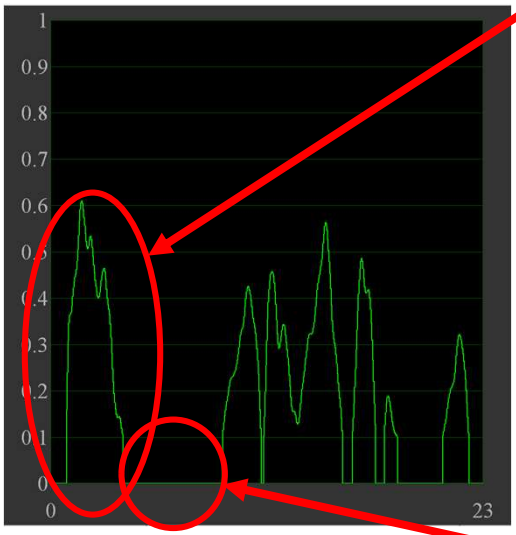
Fixed threshold applied to Motion Index



Motion Index after segmentation

147

Unitizing by thresholding



Motion Index after segmentation

Movement unit

Pause

148

Output: mid-level features

casa Paganini infomus

- For each movement unit (possibly a gesture), the T time-series received as input are processed.
- N_1, N_2, … N_T scalar values are computed from time-series 1, 2, … T respectively: these are features describing the behavior of the time-series during a movement unit.
- At the end of the movement unit, a vector

$$\boldsymbol{g} = [g_{11}, \dots, g_{1N_1}, g_{21}, \dots, g_{2N_2}, \dots, g_{T1}, \dots, g_{TN_T}]$$

is returned, consisting of mid-level features whose values are expected to characterize what was observed within the unit (i.e., they are annotated scores for the movement unit).

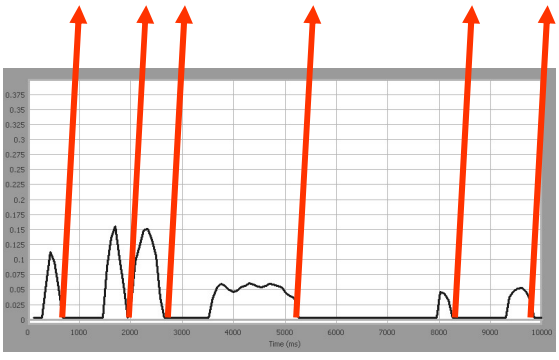
149

Output: mid-level features

casa Paganini infomus

- We move from synchronous to asynchronous data.

Motion Index	0.1	0.15	0.15	0.05	0.05	0.07
Contraction Index	0.3	0.9	0.3	0.5	0.7	0.1
Asymmetry Index	0.8	0.9	0.8	0.1	0.2	0.3



An example:

$\boldsymbol{g} = [avg(MI), avg(CI), avg(SI)]$

150

Mid-level features

casa Paganini infomus

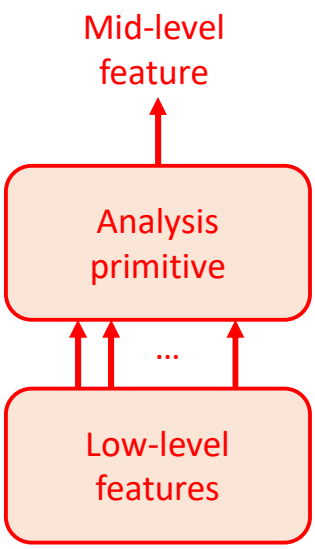
- Two major approaches for computing them:
  - Application of analysis primitives to one or many low-level features.
  - Direct computation of specifically defined features.
- Analysis primitives are unary, binary, or *n*-ary operators that summarize with one or more values the temporal dynamics of a feature in a movement unit.

151

Analysis primitives

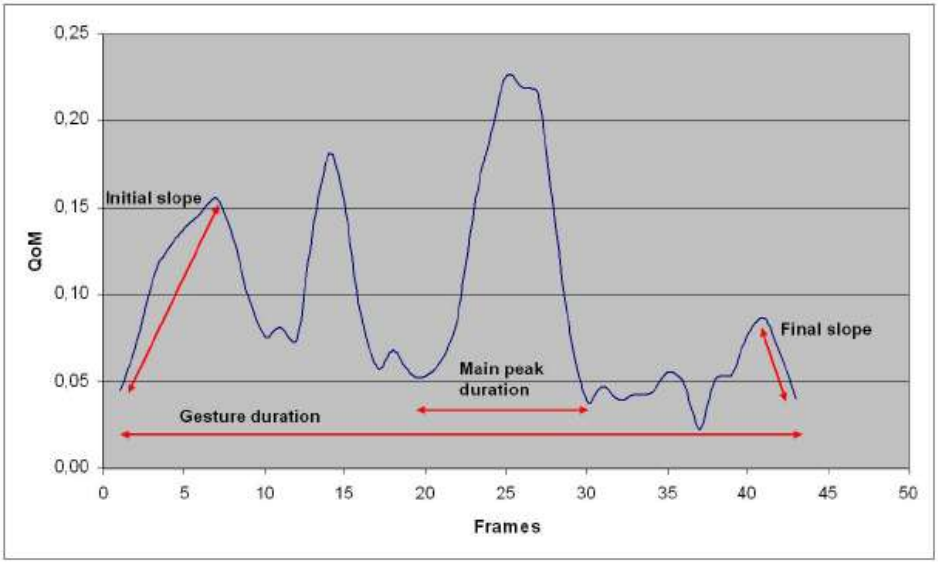
casa Paganini infomus

- Statistical operations
  - E.g., average, variance, skewness, kurtosis.
- Salient events and shape
  - E.g., slope, peaks and valleys.
- Complexity
  - E.g., sample entropy.
- Intrapersonal synchronization
  - E.g., recurrence quantification analysis.
- Causality and interdependence
  - E.g., Granger causality.



152

Analysis primitives: examples



153

Mid-level features

- More complex features can be computed from the simplest ones. A couple of examples:
  - Impulsivity: an impulsive movement is a sudden movement, which is not prepared, i.e., it is a movement displaying high acceleration and performed with no premeditation (Niewiadomski et al., 2015).
  - Fluidity: a fluid movement is smooth, and energy propagates along the kinematic chains of the body. I.e., there is an efficient propagation of movement along the kinematic chains, and dissipation of energy is minimized (Piana et al., 2016).

154

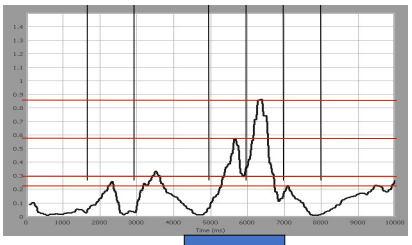
Mid-level features

Fluid vs. rigid movement

Impulsive movement

155

Symbolic representations



RHYTHM

Movements emphasized through pauses

Languages for Choreography

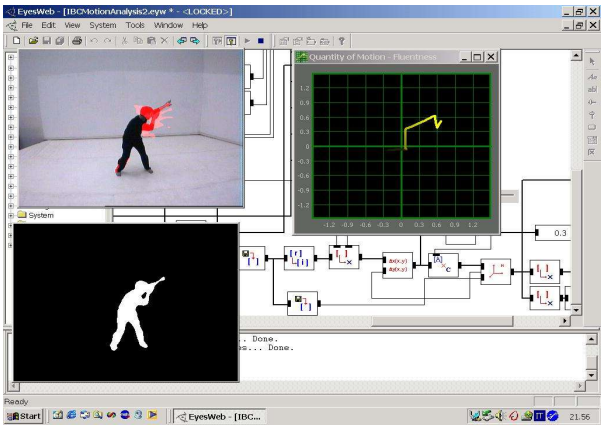
motion_unit(26 , 8 , 0.0325 , 0.0243 , 2 , 0.039 , -0.407 , 0.342).
pause(34 , 3, 0.0321, 0.3435).
motion_unit(37 , 8 , 0.0462, 0.0495 , 6 , 0.1112 , 0.297 , 0.4908).

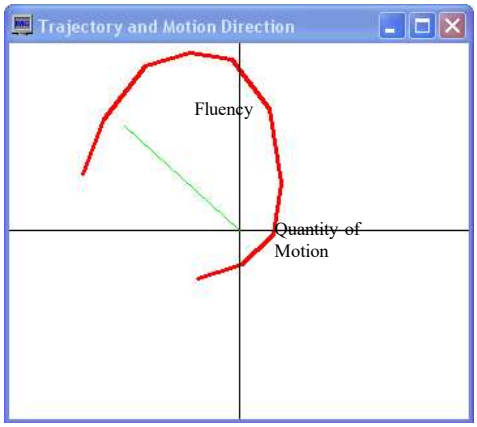
156

Dimensional representations

casa Paganini infomus

- Movement units as points/trajectories in spaces.

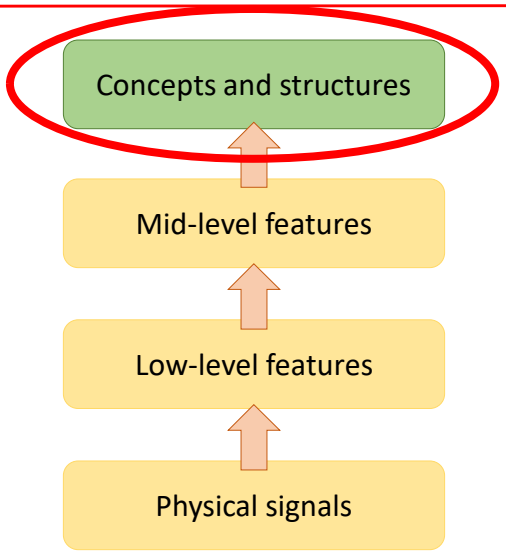




157

A conceptual framework

casa Paganini infomus



158

Gesture recognition

casa Paganini infomus

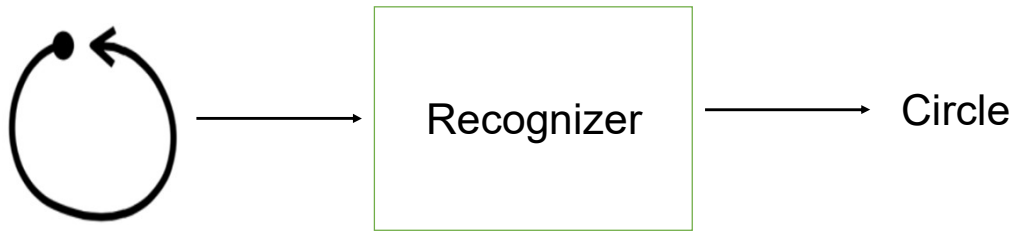
- Interpretation of human gestures via algorithms. Current focuses in the field include recognition of shapes, postures, and human behaviors (e.g., gait, emotions).
- Two major tasks for gesture recognition systems:
  - Which gesture was performed?
  - How was the gesture performed?
- Traditional and consolidated gesture recognition systems mainly deal with the first task; systems for expressive gesture processing mainly deal with the second one.

159

Gesture recognition

casa Paganini infomus

- Gesture recognizer: a system able to take an unknown input gesture and classify it as being one element of a predefined set of gestures (vocabulary).

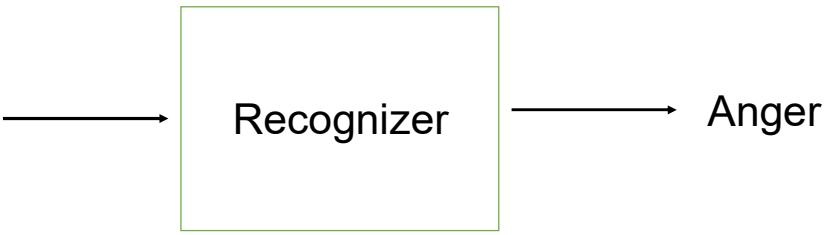


160

Gesture recognition

casa Paganini infomus

- When dealing with expressive gesture, the recognizer associates a set of movement features to high-level structures such as, for example, emotions.



161

Approaches

casa Paganini infomus

- Template-based approaches: they align the input sequence with a reference one (template), and compute distance between input gesture and pre-recorded gestures. Examples:
  - Rubine's algorithm (Rubine, 1991).
  - The $ family ($1, $N) (Wobbrock et al., 2007).
  - Dynamic Time Warping (DTW).
- Machine learning approaches: they learn to identify similar patterns by training, and take decision based on the built model. Examples: kNN, HMM, SVM, ANN, …

162

$1 algorithm

casa Paganini infomus

- An algorithm for recognition of 2D single-stroke gestures.
- Each gesture is represented as a sequence of M 2D points $\boldsymbol{G} = \Big(\big(g_x(t_0), g_y(t_0)\big), \dots, \big(g_x(t_{M-1}), g_y(t_{M-1})\big)\Big)$.
- Comparison is performed with respect to K template gestures, consisting of N 2D points each:

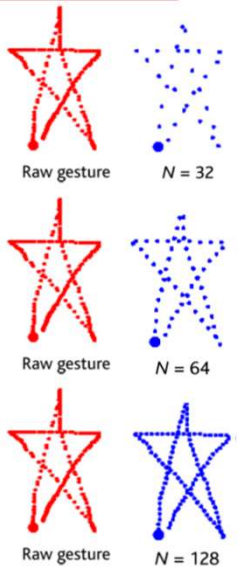
$$\boldsymbol{T}_1 = \Big(\big(t_x^1(t_0), t_y^1(t_0)\big), \dots, \big(t_x^1(t_{N-1}), t_y^1(t_{N-1})\big)\Big)$$
$$\boldsymbol{T}_2 = \Big(\big(t_x^2(t_0), t_y^2(t_0)\big), \dots, \big(t_x^2(t_{N-1}), t_y^2(t_{N-1})\big)\Big)$$
$$\dots$$
$$\boldsymbol{T}_K = \Big(\big(t_x^K(t_0), t_y^K(t_0)\big), \dots, \big(t_x^K(t_{N-1}), t_y^K(t_{N-1})\big)\Big)$$

163

$1 algorithm: step 1

casa Paganini infomus

- Resampling the point path: resampling gestures such that the path defined by their original M points is defined by N equidistantly spaced points. Generally, N= 64. Steps:
  - Compute length L of the M-points path.
  - Compute increment I as L / (N – 1).
  - Add a new point (coordinates computed through linear regression) whenever the distance between 2 points of the original path is greater than I.



Picture from (Wobbrock et al. 2007)

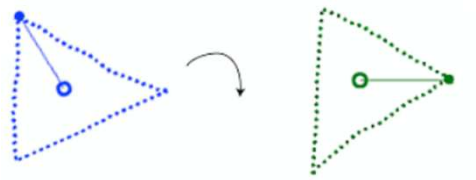
164

$1 algorithm: step 2

casa Paganini infomus

- Rotate once based on the gesture's indicative angle: this is defined as the angle formed between the centroid of the gesture and the gesture's first point. Steps:
  - Compute the centroid of the gesture.
  - Compute the angle between the first point of the trajectory, the centroid, and the horizontal line.
  - Rotate the points of this angle.

Rotation is performed so that the gesture's indicative angle is 0°. Picture from (Wobbrock et al. 2007).

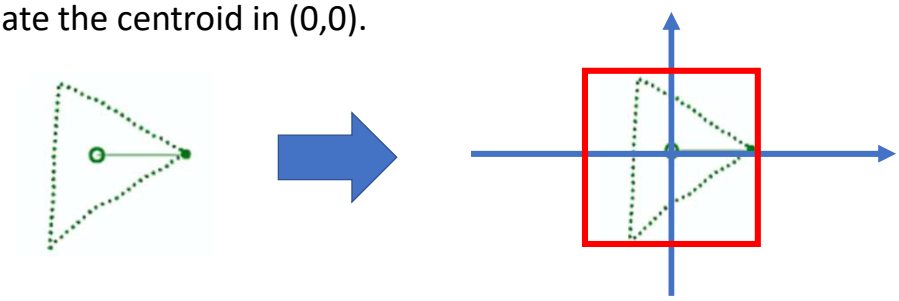


165

$1 algorithm: step 3

casa Paganini infomus

- Scale and translate: the gesture is scaled to a reference square (non-uniform scaling). After scaling, the gesture is translated to a reference point (centroid in (0,0)). Steps:
  - Compute the gesture's bounding box.
  - Scale to a reference square having a pre-fixed size.
  - Translate the centroid in (0,0).



166

$1 algorithm: step 4

casa Paganini infomus

- Recognition: a candidate gesture $\boldsymbol{G}$ is compared with each template gesture $\boldsymbol{T}_i$. Steps:
  - Compute the path-distance d_i between $\boldsymbol{G}$ and $\boldsymbol{T}_i$ (averaged distance between the points).

$$d_i = \frac{\sum_{j=1}^{N} \sqrt{\left(g_x(t_j) - t_x^i(t_j)\right)^2 + \left(g_y(t_j) - t_y^i(t_j)\right)^2}}{N}$$

  - The template $\boldsymbol{T}_{\hat{\imath}}$ obtaining the minimum path-distance to $\boldsymbol{G}$ is taken as the output of the recognition. That is:

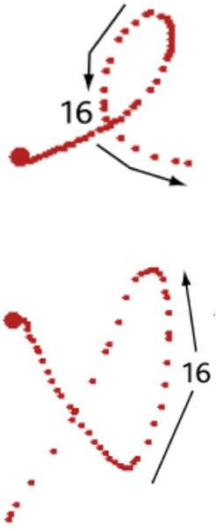
$$\hat{\imath} = \underset{i=1,\ldots,K}{\operatorname{argmin}}\, d_i$$

167

$1 algorithm

casa Paganini infomus

- Pros:
  - Resilient to variations in sampling due to movement speed or sensing device.
  - Able to support rotation and scale.
  - Simple to implement and fast.
  - Supporting developers and end-users to teach it new gestures with only one example.
- Limitations:
  - No way to distinguish a rectangle by a square.
  - No way to distinguish an ellipse by a circle.
  - No way to distinguish the orientation of an arrow.



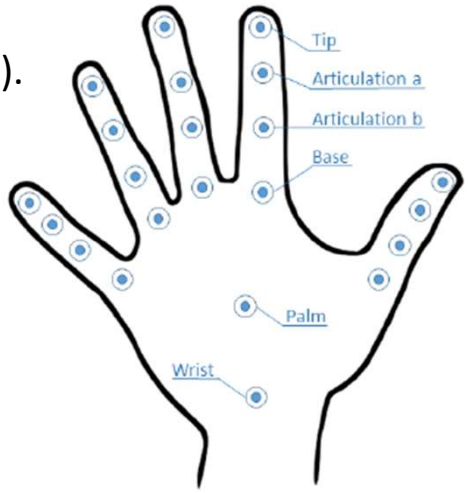
Picture from (Wobbrock et al. 2007)

168

A machine learning approach

casa Paganini infomus

- 3D hand gesture recognition from skeletal data (De Smedt et al, 2016).
- Gestures are described as a time series of hand skeletons.
- The trajectory of each key-point $\boldsymbol{p}^k$ in a hand skeleton is:
  $\mathbf{P}^k = (\boldsymbol{p}^k(t_0), \boldsymbol{p}^k(t_1), \dots, \boldsymbol{p}^k(t_N))$
  with:
  $\boldsymbol{p}^k(t_i) = [x^k(t_i), y^k(t_i), z^k(t_i)]^{\mathrm{T}}$



De Smedt, Q., Wannous, H., and Vandeborre, J., 2016. Skeleton-Based Dynamic Hand Gesture Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 1206-1214.

169

Feature extraction

casa Paganini infomus

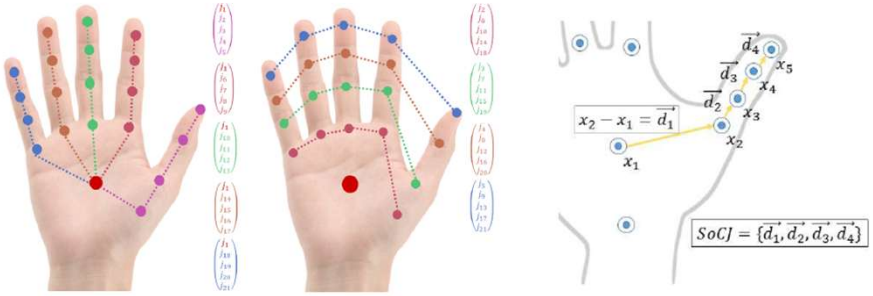
- Hand gestures are modeled with motion and shape features.
- Motion features:
  - Direction: $\boldsymbol{d}(t_i) = \dfrac{\boldsymbol{p}^{palm}(t_i) - \boldsymbol{p}^{palm}(t_{i-c})}{\|\boldsymbol{p}^{palm}(t_i) - \boldsymbol{p}^{palm}(t_{i-c})\|}$
  - Rotation: $\mathbf{r}(t_i) = \dfrac{\boldsymbol{p}^{palm}(t_i) - \boldsymbol{p}^{wrist}(t_i)}{\|\boldsymbol{p}^{palm}(t_i) - \boldsymbol{p}^{wrist}(t_i)\|}$
- Direction vectors are aggregated in the M_{dir} direction matrix.
- Rotation vectors are aggregated in the M_{rot} rotation matrix.
- Both are $N_f \times 3$ matrices: each row is a direction or rotation vector and N_f is the number of frames.

170

Feature extraction

casa Paganini infomus

- Shape features:
  - Normalization (to remove personal differences in hand size).
  - Shape of Connected Joints (SoCJ) computation: 9 SoCJ are computed at each frame and grouped together.
  - Fisher Vector representation of SoCJ.

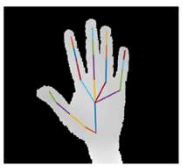


171

Experiments and results

casa Paganini infomus

- 3D hand skeletal data (22 key-points) from depth maps captured with an Intel RealSense Camera (30fps, 640x480).
  - 14 types of gestures (2800 gestures)
  - 2 ways (one vs. more fingers)
  - 28 participants
  - 1-10 repetitions per participant
  - From 20 to 50 frames duration



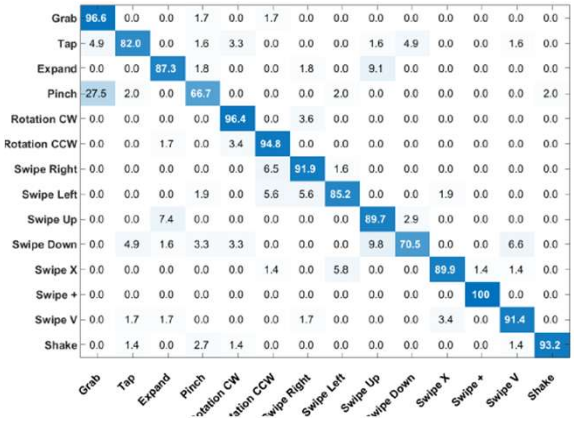
Gesture	Label	Tag name
Grab	Fine	G
Expand	Fine	E
Pinch	Fine	P
Rotation CW	Fine	R-CW
Rotation CCW	Fine	R-CCW
Tap	Coarse	T
Swipe Right	Coarse	S-R
Swipe Left	Coarse	S-L
Swipe Up	Coarse	S-U
Swipe Down	Coarse	S-D
Swipe X	Coarse	S-X
Swipe V	Coarse	S-V
Swipe +	Coarse	S-+
Shake	Coarse	Sh

172

Experiments and results

casa Paganini infomus

- SVM classifier with linear kernel (high-dimensional data).
- Leave-One-Subject-Out (LOSO) evaluation strategy.

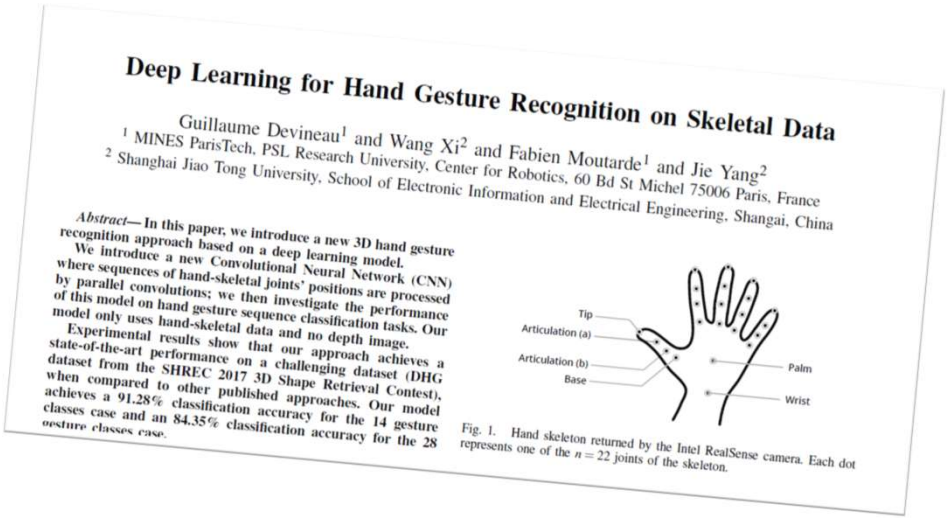


Method	14 gestures (%)	28 gestures (%)
Guerry *et al.*	82.90	71.90
De Smedt *et al.* [DSWV16]	88.24	81.90
Ohn-Bar *et al.* [OBT13]	83.85	76.53
Oreifej *et al.* [OL13]	78.53	74.03
Devanne *et al.* [DWB*15]	79.61	62.00

173

A deep learning version

casa Paganini infomus



Deep Learning for Hand Gesture Recognition on Skeletal Data

Guillaume Devineau[1] and Wang Xi[2] and Fabien Moutarde[1] and Jie Yang[2]
[1] MINES ParisTech, PSL Research University, Center for Robotics, 60 Bd St Michel 75006 Paris, France
[2] Shanghai Jiao Tong University, School of Electronic Information and Electrical Engineering, Shangai, China

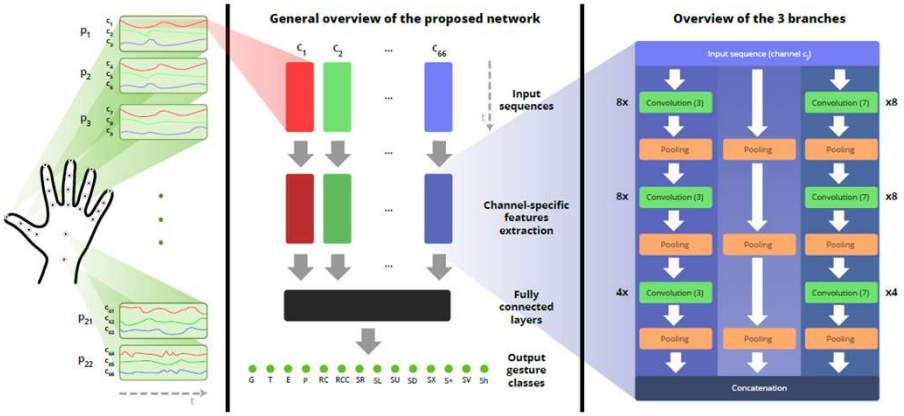
Abstract— In this paper, we introduce a new 3D hand gesture recognition approach based on a deep learning model. We introduce a new Convolutional Neural Network (CNN) where sequences of hand-skeletal joints' positions are processed by parallel convolutions; we then investigate the performance of this model on hand gesture sequence classification tasks. Our model only uses hand-skeletal data and no depth image. Experimental results show that our approach achieves a state-of-the-art performance on a challenging dataset (DHG dataset from the SHREC 2017 3D Shape Retrieval Contest), when compared to other published approaches. Our model achieves a 91.28% classification accuracy for the 14 gesture classes case and an 84.35% classification accuracy for the 28 gesture classes case.

Fig. 1. Hand skeleton returned by the Intel RealSense camera. Each dot represents one of the $n = 22$ joints of the skeleton.

Devineau, G., Moutarde, F., Xi, W., and Yang, J., 2018. Deep Learning for Hand Gesture Recognition on Skeletal Data, 2018. 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), 106-113.

174

Architecture



3 branches:
2 branches for feature extraction (high vs. low resolution), 1 residual branch

- Preprocessing: data resampled (linear interpolation) to get fixed-length gestures (100 time-steps).

175

Experiments and results

- 2800 gestures: 1960 for training (70%), 840 for testing (30%).
- Two classification tasks: 14 gestures vs. 28 gestures.
- No information about the evaluation protocol (LOSO?).

	G	T	E	P	RC	RCC	SR	SL	SU	SD	SX	S+	SV	Sh
G	94.8	1.7	0.0	1.7	1.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
T	11.5	77.0	0.0	3.3	4.9	0.0	0.0	0.0	0.0	3.3	0.0	0.0	0.0	0.0
E	0.0	5.5	90.9	0.0	0.0	0.0	1.8	0.0	0.0	0.0	0.0	0.0	0.0	1.8
P	15.7	0.0	0.0	78.4	3.9	2.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RC	0.0	1.8	0.0	0.0	98.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RCC	3.4	6.9	0.0	1.7	5.2	77.6	0.0	1.7	0.0	1.7	0.0	0.0	0.0	1.7
SR	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
SL	0.0	0.0	0.0	0.0	7.4	0.0	3.7	88.9	0.0	0.0	0.0	0.0	0.0	0.0
SU	2.9	2.9	10.3	0.0	0.0	0.0	1.5	0.0	79.4	1.5	0.0	0.0	1.5	0.0
SD	3.3	0.0	0.0	0.0	3.3	0.0	0.0	0.0	0.0	91.8	0.0	0.0	1.6	0.0
SX	0.0	2.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	89.9	1.4	5.8	0.0
S+	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0
SV	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0
Sh	0.0	8.6	2.9	0.0	12.9	0.0	2.9	0.0	1.4	0.0	0.0	0.0	0.0	71.4

In this confusion matrix each row represents the real class of performed gestures while each column represents the predicted class of the gestures.

	Ours			De Smedt *et al.*			Difference
Gesture	Precision	Recall	F_1-score	Precision	Recall	F_1-score	F_1-score
G	72.4%	94.8%	**82.1%**	67.5%	57.0%	61.8%	20.3%
T	71.2%	77.0%	74.0%	85.2%	87.0%	**86.1%**	-12.1%
E	84.7%	90.9%	**87.7%**	84.8%	87.0%	85.9%	1.8%
P	90.9%	78.4%	**84.2%**	52.1%	61.0%	56.2%	28.0%
RC	69.2%	98.2%	**81.2%**	80.0%	77.5%	78.8%	2.5%
RCC	97.8%	77.6%	86.5%	90.9%	85.5%	**88.1%**	-1.6%
SR	91.2%	100.0%	**95.4%**	85.1%	92.5%	88.6%	6.7%
SL	98.0%	88.9%	**93.2%**	78.4%	85.5%	81.8%	11.4%
SU	98.2%	79.4%	**87.8%**	89.3%	85.5%	87.4%	0.4%
SD	93.3%	91.8%	**92.6%**	80.8%	88.0%	84.3%	8.3%
SX	100.0%	89.9%	**94.7%**	95.8%	85.0%	90.1%	4.6%
S+	98.3%	100.0%	**99.1%**	90.2%	98.5%	94.1%	5.0%
SV	90.6%	100.0%	**95.1%**	93.2%	92.0%	92.6%	2.5%
Sh	96.2%	71.4%	82.0%	88.6%	81.0%	**84.7%**	-2.7%

176

Expressive gesture: an example

casa Paganini infomus

- A system for automatic classification of basic emotions in micro-dances (Camurri et al., 2004):
  - Extraction of expressive features.
  - Explorative analysis on the extracted features.
  - Automatic classification.
  - Results of automatic classification compared with spectators' ratings of the same dances.

Photo by Robert Collins on Unsplash

177

Expressive gesture: an example

casa Paganini infomus

- Dataset: five dancers performed four times the same choreography, each time with a different expressive intention, corresponding to four basic emotions: Anger, Fear, Grief, Joy.
  The dataset thus consists of 20 dance performances.

178

Expressive gesture: features

casa Paganini infomus

Basic Emotion	Expressive Cues
Anger	Short duration of time Frequent tempo changes, short stops between change Movements reaching out from body centre Dynamic and high tension in the movement Tension builds up and then "explodes"
Fear	Frequent tempo changes Long stops between changes Movements kept close to body centre Sustained high tension in movements
Grief	Long duration of time Few tempo changes, "smooth tempo" Continuously low tension in the movements
Joy	Frequent tempo changes Longer stops between changes Movements reaching out from body centre Dynamic tension in movements Changes between high and low tension

Table from (Camurri, Lagerlöf, Volpe, 2003).

179

Expressive gesture: features

casa Paganini infomus

- 18 (standardized) features:
  - Motion Index and its first and second order statistics.
  - Contraction Index and its first and second order statistics.
  - Upward Movement.
  - Directness Index.
  - Kinematic features.
- Dances unitized in 334 motion units using a fixed threshold applied to the Motion Index.
- Explorative analysis applied to the computed features.

180

Expressive gesture: classification

casa Paganini infomus

- Five decision tree models (85% training set, 15% test set randomly selected, stratified over the four classes).
- Average accuracy
  - Spectators: 56%.
  - Automatic classification: 36%.
    (64% if we take as reference spectators' performance).
- Best model: 40% (71% relative).
- Chance level: 25%.
- The performance of the automatic classifier is in between chance level and the performance of human observers.

181

Expressive gesture: classification

casa Paganini infomus

- Confusion matrix for the test set for the best model:

Class	Total	%Correct	%Error	Anger	Fear	Grief	Joy
Anger	12	41.7	58.3	5	3	0	4
Fear	13	30.8	69.2	6	4	2	1
Grief	12	41.7	58.3	2	0	5	5
Joy	13	46.1	53.8	4	0	3	6

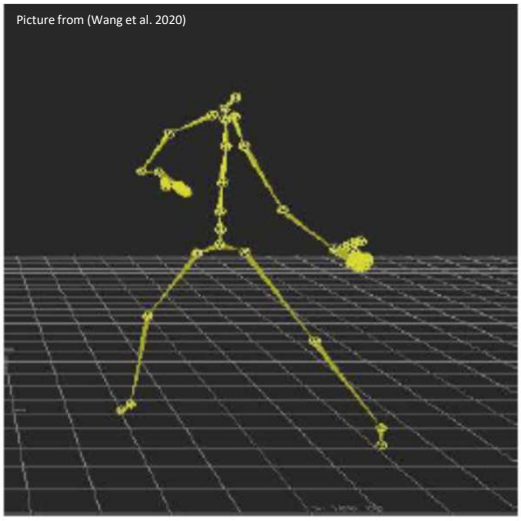
- Removed facial expressions may explain the quite low recognition rate for spectators (56%).
- The gap between automatic recognition and spectators' ratings could be due to neglected temporal aspects.

182

Another example

casa Paganini infomus

- Wang et al. (2020) proposed a hybrid deep architecture to recognize 7 emotions (fear, anger, boredom, excitement, joy, relaxation, sadness) from Laban's features.
- Input: motion capture data (3D positions and Euler angles) of 54 body landmarks (6 people recorded).



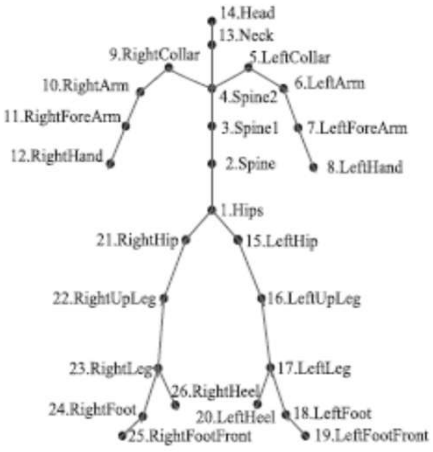
Wang, S., Li, J., Cao, T., Wang, H., Tu, P., and Li, Y., 2020. Dance Emotion Recognition Based on Laban Motion Analysis Using Convolutional Neural Network and Long Short-Term Memory. IEEE Access, 8, 124928-124938.

183

Another example

casa Paganini infomus

- Features:
  - Inspired by Laban Motion Analysis, focus on spatial orientation, limb structure, and Effort.
  - 52 distances between joints, 54 speeds, 54 accelerations.
  - Unitizing: interval-coding (averaged features over 1s).

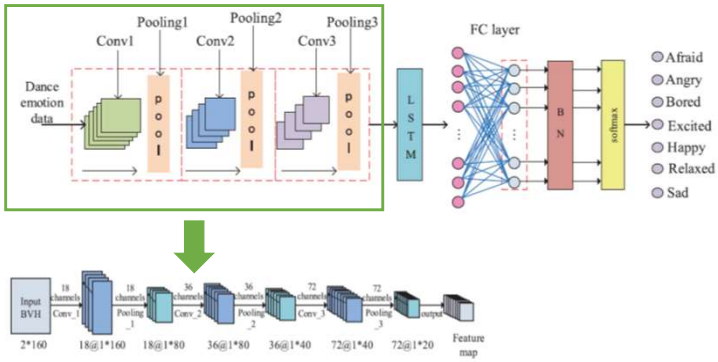


Picture from (Wang et al. 2020)

184

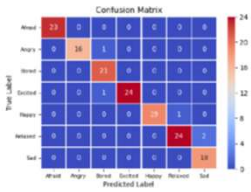
Another example

casa Paganini infomus



Architecture

Results



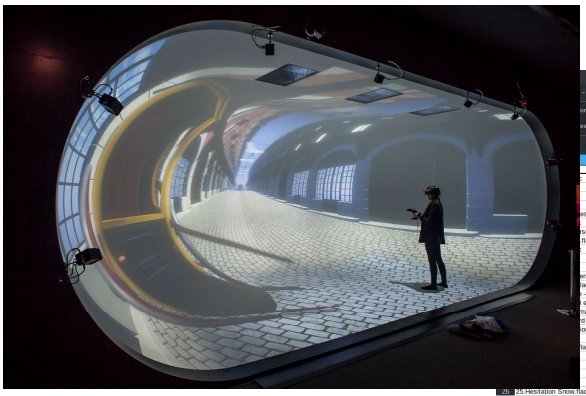
	CNN-LSTM	CNN	LSTM
Afraid	0.94	0.88	0.90
Angry	0.93	0.86	0.89
Bored	0.95	0.90	0.90
Excited	0.95	0.89	0.88
Happy	0.95	0.90	0.87
Relaxed	0.95	0.88	0.82
Sad	0.95	0.86	0.85
Average	0.95	0.88	0.87

185

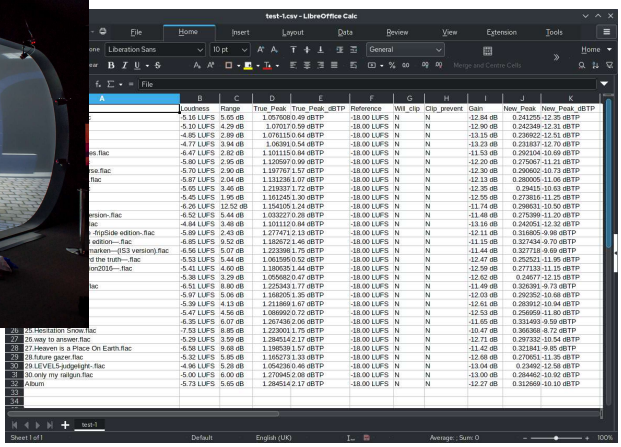
2. Designing a Multimodal System

1

Is a multimodal system needed?



Source: https://blog.laval-virtual.com/en/glossary/cave/



By Software: The Document Foundation, Alessandro Ghedini, Matthias C. Hormann, and contributors. Screenshot: VulcanSphere - Self-taken; derivative work, MPL 2, https://commons.wikimedia.org/w/index.php?curid=113811778

2

Motivations for multimodality

1. Human-human communication is multimodal - Unimodal communication is an artifact of communications technology
2. Input and output by the most effective means - "*Certain tasks and functions cry out for particular modalities*" (Rudnicky and Hauptmann 1992)
3. Adapting to the environment - Physical and social
4. Migration of human-computer interaction away from the desktop - Smartphones, PDAs, Wall-size displays
5. Task performance and user preference - Oviatt 1996, 1997, Nishimoto et al 1997, Hauptmann 1989
6. Error handling - Switching modes to avoid error spirals (Oviatt and van Gent 1996) - Mutual compensation (Oviatt 1999, Bangalore and Johnston 2000)

Table by Michael Johnston, AT&T Labs-Research, 2006.

Johnston, M., 2019. Multimodal integration for interactive conversational systems. In S. Oviatt, B. Schuller, P. R. Cohen, D. Sonntag, G. Potamianos, and A. Krüger (Eds.), The Handbook of Multimodal-Multisensor Interfaces. Association for Computing Machinery and Morgan & Claypool, New York, NY, USA, 21-76 (see section 1.2).

3

Human-human communication

- Speech is accompanied by gesture and body movement.
- Gaze and eye contact is very important.
- Face expression is very significant.
- Prosody is a crucial component of speech.

4

Example: a famous interview

Dialogue transcripts

- "… Guy Kewney is the editor of the technology website News Wireless. Hello, good morning to you."
- "Good morning."
- "Were you surprised by this verdict today?"
- "I'm very surprised to see this verdict to come on me. Because I was not expecting that. When I came, they told me something else and I'm coming. And they told me something else. Big surprise any way."
- "A big surprise..."
- "Exactly."
- "Yeah yeah. With regard to the cost that is involved. Do you think more people will be downloading online?"
- "Actually if you can go everywhere, you gonna see people downloading through the internet and the websites. [...]

5

Example: a famous interview

- Let's listen to the audio.

6

Example: a famous interview

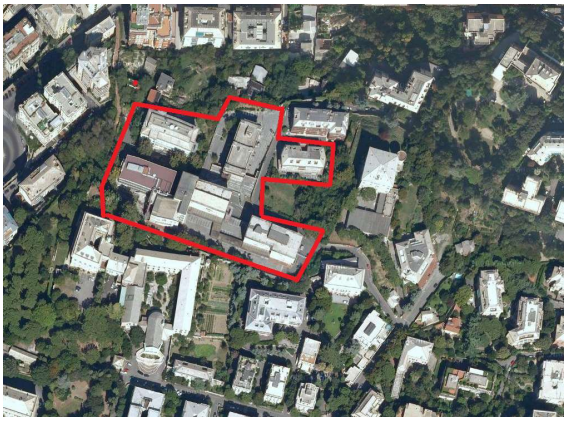
Full video

Guy Goma on BBC News 24 on Monday 8 May 2006.

7

Most effective means

- Certain kinds of content are most easily expressed in specific non-verbal modalities.
- But some information is better suited to verbal communication, e.g., “the left bank of the river”.



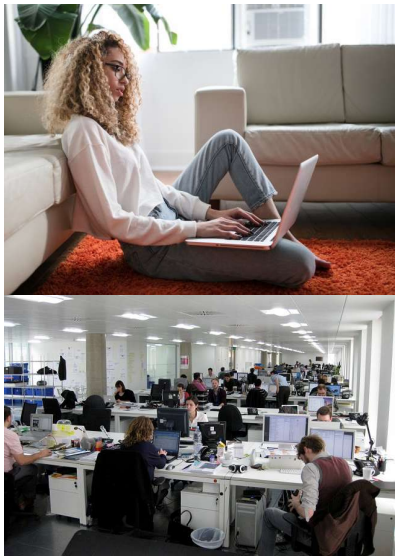
Example: drawing the borders of a region

8

Adapting to the environment

casa Paganini infomus

- Multimodal systems enable rapid adaptation to changing environment, by switching to the most suitable modality or by complementing different modalities.
- Adaptation to changes in the physical environment: e.g., adaptation to ambient noise, darkness, brightness, and so on.
- Adaptation to changes in the social environment: e.g., single user vs. multiple users, at home vs. at work, and so on.



9

Performance and preference

casa Paganini infomus

- Many empirical studies show that multimodal interfaces improve task performance and are preferred by users over unimodal interfaces. For example:
  - Better than unimodal speech for map-based tasks (Oviatt, 1996);
  - Faster than GUI for map-based tasks (Cohen et al., 1998);
  - Faster than GUI for drawing applications (Nishimoto et al., 1995);
  - User preference for speech and gesture in object manipulation tasks (Hauptmann, 1989).

Oviatt, S., 1996. Multimodal interfaces for dynamic interactive maps. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'96), 95-102.

Cohen, P. R., Johnston, M., McGee, D., Oviatt, S. L., Clow, J., and Smith, I., 1998. The efficiency of multimodal interaction: a case study. In Proceedings 5th International Conference on Spoken Language Processing (ICSLP-1998), paper 0571.

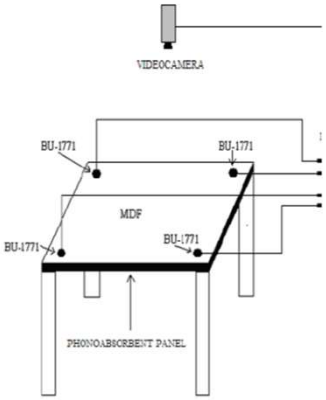
Nishimoto, T., Shida, N., Koayashi, T., and Shirai, K., 1995. Improving human interface drawing tool using speech, mouse and key-board. In Proceedings 4th IEEE International Workshop on Robot and Human Communication, 107-112.

Hauptmann, A. G., 1989. Speech and gestures for graphic image manipulation. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'89), 241-245.

10

Error handling

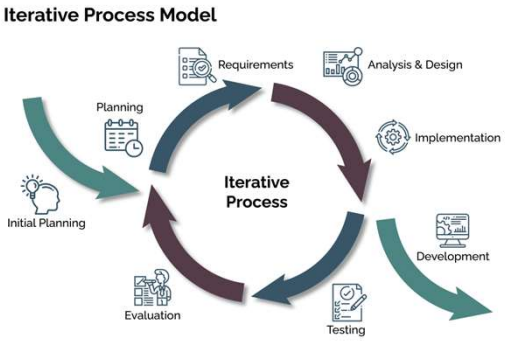
- Multimodal interfaces can reduce errors:
  - Mode switching: use of an alternate modality to escape error spirals in a modality.
  - Cross-modal compensation: use of information from a modality to compensate errors in another modality (e.g., compensation of audio information with visual information in touch localization in Tangible Acoustic Interfaces).
  - Multimodal confirmation: use of information from a modality for confirming results obtained by data from another modality.



11

Design tools

- The design process usually adopts the same procedures commonly employed in HCI.
- Conceptual tools are available:
  - Design guidelines
  - Models for modality choice
  - Models for multimodal combination
  - Frameworks for multimodal systems



12

Guidelines (Reeves et al., 2004)

- Be consistent – in system output, presentation and prompts, enabling shortcuts, state switching, …
- Provide good error prevention and error handling, make functionality clear and easily discoverable.
- Multimodal systems should be designed for the broadest range of users and contexts of use.
  Support the best modality or combination of modalities anticipated in changing environments (e.g., office vs. car).
- Designers should take care to address privacy and security issues in multimodal systems.
  E.g., provide non-speech alternatives in a public context.

13

Guidelines (Reeves et al., 2004)

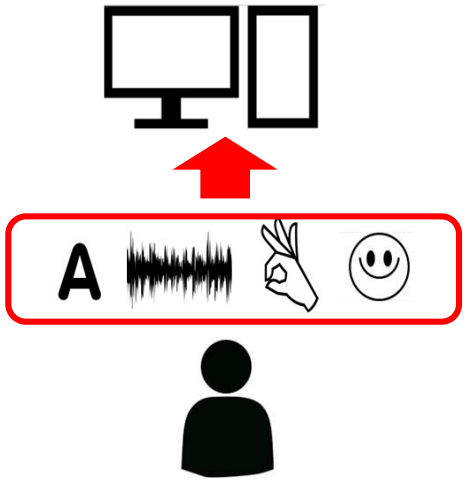
- Maximize human cognitive and physical abilities, based on an understanding of users' human information processing abilities and limitations.
- Modalities should be integrated in a manner compatible with user preferences, context, and system functionality.
  E.g., match the output to acceptable user input style, such as constrained grammar or unconstrained natural language.
- Multimodal interfaces should adapt to different users, as well as different contexts of use. Capture individual differences in a user profile and use it for interface settings.

Reeves, L. M., Lai, J., Larson, J. A., Oviatt, S., Balaji, T. S., Buisine, S., Collings, P., Cohen, P., Kraal, B., Martin, J-C. McTear, M., Raman, T. V., Stanney, K. M., Su, H., and Wang, Q. Y., 2004. Guidelines for multimodal user interface design. Communications of the ACM 47, 1, 57-59.

14

How do users choose modalities?

- Modality combination: a set of input modalities (sometimes just one!) that a user adopts to perform a specific task with a multimodal system.
- Modality choice: the selection of a modality combination to be adopted in a specific context. Such a selection can be conscious / unconscious.



15

Modalities

- Overview of sensory modalities at the neurophysiological level.

Sensory modality	Form of energy	Receptor organ	Receptor cell
Chemical (internal)			
blood oxygen	O_2 tension	carotid body	nerve endings
glucose	carbohydrate oxidation	hypothalamus	gluco-receptors
pH (cerebrospinal fluid)	ions	medulla	ventricle cells
Chemical (external)			
taste	ions & molecules	tongue & pharynx	taste bud cells
smell	molecules	nose	olfactory receptors
Somatic senses			
touch	mechanical	skin	nerve terminals
pressure	mechanical	skin & deep tissue	encapsulated nerve endings
temperature	thermal	skin, hypothalamus	peripheral & central
pain	various	skin & various organs	nerve terminals
Muscle sense, kinesthesia			
muscle stretch	mechanical	muscle spindles	nerve terminals
muscle tension	mechanical	tendon organs	nerve terminals
joint position	mechanical	joint capsule & ligaments	nerve terminals
Sense of balance			
linear acceleration	mechanical	sacculus/utriculus	hair cells
angular acceleration	mechanical	semicircular canal	hair cells
Hearing			
	mechanical	cochlea	hair cells
Vision			
	light	retina	photoreceptors

16

Modalities

casa Paganini infomus

- In multimodal interactive systems (Blattner and Glinert, 1996):

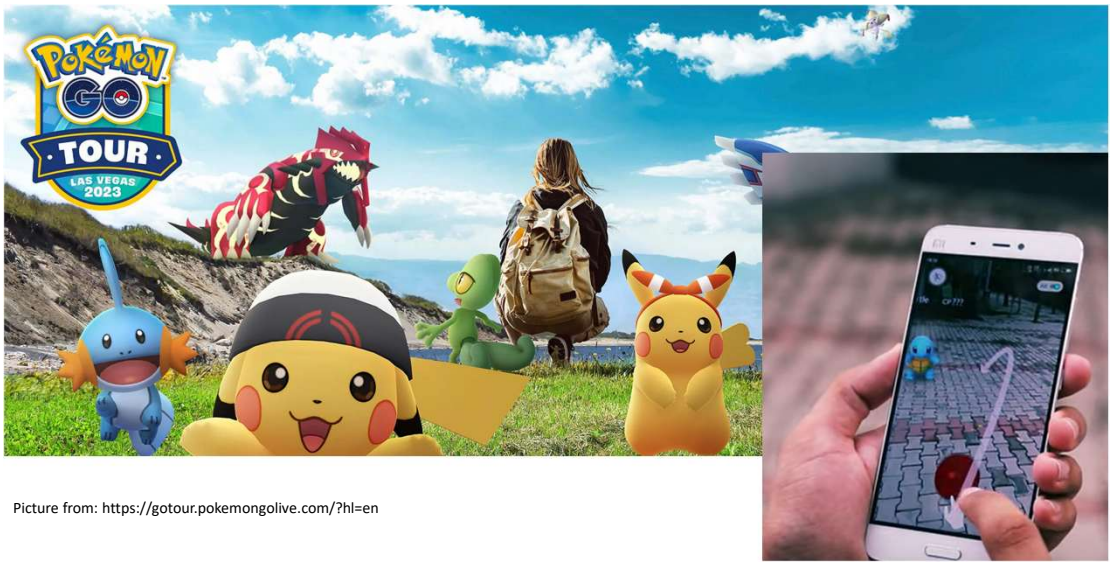
Modality	Example
Visual	Face location
	Gaze
	Facial expression
	Lipreading
	Face-based identity (and other user characteristics such as age, sex, race, etc.)
	Gesture (head/face, hands, body)
	Sign language
Auditory	Speech input
	Non-speech audio
Touch	Pressure
	Location and selection
	Gesture
Other sensors	Sensor-based motion capture

Adapted from: Blattner, M. M., and Glinert, E. P., 1996. Multimodal integration. IEEE MultiMedia, 3, 4, 14-24.

17

The Pokemon Go case

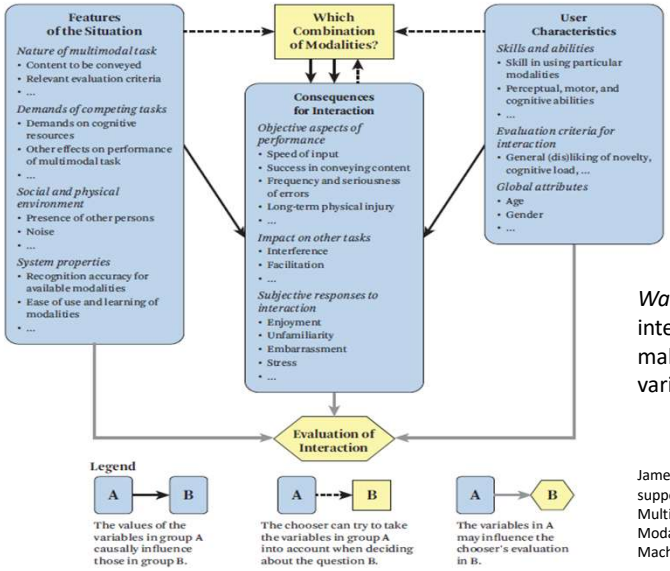
casa Paganini infomus



Picture from: https://gotour.pokemongolive.com/?hl=en

18

Variables for modality choice



Picture from: Jameson and Kristensson, 2017

Warning: "The diagram is by no means intended to suggest that users actually make their choices by considering so many variables" (Jameson and Kristensson, 2017).

Jameson, A., and Kristensson, P.O., 2017. Understanding and supporting modality choices. In The Handbook of Multimodal-Multisensor Interfaces: Foundations, User Modeling, and Common Modality Combinations - Volume 1. Association for Computing Machinery and Morgan & Claypool, 201–238.

19

ASPECT and ARCADE models

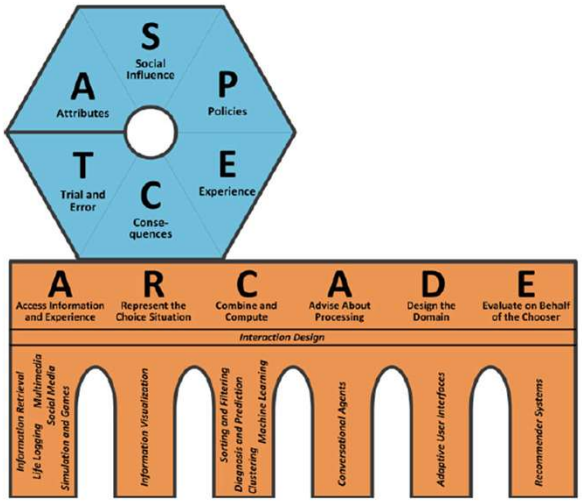
- ASPECT and ARCADE are two models firmly rooted on psychological research.
- They concern how people make choices in general, but they can be specifically applied to modality choice.

ASPECT covers the variety of ways in which people make choices

ARCADE summarizes the ways in which it is possible to help people make a better choice

20

ASPECT and ARCADE models



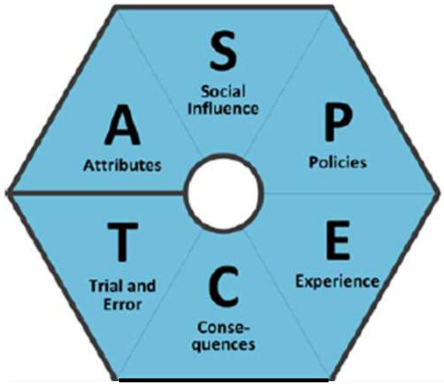
Picture from: Jameson et al., 2015.

Jameson, A., Willemsen, M., Felfernig, A., de Gemmis, M., Lops, P., Semeraro, G., and Chen, L., 2015. Human Decision Making and Recommender Systems. In Ricci, F., Rokach, L., Shapira, B. (eds) Recommender Systems Handbook. Springer, Boston, MA.

21

The ASPECT model

- ASPECT captures the complexity and the variety of choices by specifying six general patterns that users can apply alternatively or in combination.
- The six patterns are:
  - Consequence-based
  - Trial-and-error-based
  - Policy-based
  - Experience-based
  - Socially-based
  - Attribute-based



22

The ASPECT patterns

casa Paganini infomus

Attribute-based choice	**Consequence-based choice**
Conditions of applicability	*Conditions of applicability*
– The options can be viewed meaningfully as items that can be described in terms of attributes and levels – The (relative) desirability of an item can be estimated in terms of evaluations of its levels of various attributes	– The choices are among actions that will have consequences
Typical procedure	*Typical procedure*
– (Optional:) *C* reflects in advance about the situation-specific (relative) importance of attributes and/or values of attribute levels – *C* reduces the total set of options to a smaller *consideration set* on the basis of attribute information – *C* chooses from a manageable set of options	– *C* recognizes that a choice about a possible action can (or must) be made – *C* assesses the situation – *C* decides when and where to make the choice – *C* identifies one or more possible actions (options) – *C* anticipates (some of) the consequences of executing the options – *C* evaluates (some of) the anticipated consequences – *C* chooses an option that rates (relatively) well in terms of its consequences

Table from: Jameson et al., 2015.

23

The ASPECT patterns

casa Paganini infomus

Experience-based choice	**Socially-based choice**
Conditions of applicability	*Conditions of applicability*
– *C* has made similar choices in the past	– There is some information available about what relevant other people do, expect, or recommend in this or similar situations
Typical procedure	*Typical procedure*
– *C* applies recognition-primed decision making – or *C* acts on the basis of a habit – or *C* chooses a previously reinforced response – or *C* applies the affect heuristic	– *C* considers *examples* of the choices or evaluations of other persons – or *C* considers the *expectations* of relevant people – or *C* considers explicit advice concerning the options

Table from: Jameson et al., 2015.

24

The ASPECT patterns

casa Paganini infomus

Policy-based choice	Trial-and-error based choice
Conditions of applicability	*Conditions of applicability*
– *C* encounters choices like this one on a regular basis	– The choice will be made repeatedly; or *C* will have a chance to switch from one option to another even after having started to execute the first option
Typical procedure	*Typical procedure*
– [Earlier:] *C* arrives at a policy for dealing with this type of choice – [Now:] *C* recognizes which policy is applicable to the current choice situation and applies it to identify the preferred option – *C* determines whether actually to execute the option implied by the policy	– *C* selects an option *O* to try out, either using one of the other choice patterns or (maybe implicitly) by applying an *exploration strategy* – *C* executes the selected option *O* – *C* notices some of the consequences of executing *O* – *C* learns something from these consequences – (If *C* is not yet satisfied:) *C* returns to the selection step, taking into account what has been learned

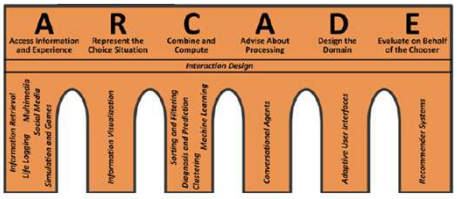
Table from: Jameson et al., 2015.

25

The ARCADE model

casa Paganini infomus

- ARCADE provides six high-level strategies for supporting choices. Such strategies are often used in combination.
- The six strategies are:
  - Access information and experience
  - Represent the choice situation
  - Combine and compute
  - Advise about processing
  - Desing the domain
  - Evaluate on behalf of the chooser



26

The ARCADE model

- Access information and experience: providing people with relevant information about the system or helping them anticipating what an experience will look like when adopting a particular choice.

  Example: a system can give a preview of what it will feel like to interact with it by means of each available modality, e.g., by making available for the users and displaying tutorial videos to them.

  Example: a system can inform the user about social examples and expectations, e.g., by showing the text of the audio messages that will be generated (and so, they can be herd by other people in the area) in case of incorrect actions performed by the user.

27

The ARCADE model

- Represent the choice situation: making users more aware of the choice of modalities …

  Example: shaping a tablet controller as a stylus will inspire user to use it as a stylus, whereas shaping it as something else will require a deeper reflection but the user could find other permissible use.

  … or subtly encouraging them to use modalities that are the most suitable in a specific context.

  Example: the system starts to talk with the user, so the user is expected to reply by using speech.

28

The ARCADE model

casa Paganini infomus

- Combine and compute: providing users with high-level relevant information (no computational details are given).

  Example: the system makes the user know through a colored led that the audio she is listening to is good. The user, however, ignores that some computation occurred (e.g., bit rate, S/N ratio).

- Advise about processing: the system suggests a procedure to choose modalities.

  Example: the system suggests the user to consider more previous experience in similar situations rather than looking at the choices made by friends.

29

The ARCADE model

casa Paganini infomus

- Design the domain: disguising that the system can work with some modalities.

  Example: in the case of an impaired user, the system suggests appropriate choices (modalities), even if other less appropriate modalities would be available in principle.

- Evaluating on behalf of the chooser: the system is smart enough to evaluate which modalities are the most suitable for a specific user.

  Example: the system assesses that the user is visually impaired and automatically switches to the auditory modality.

30

How do systems use modalities?

- Interaction and combination of modalities occur according to complex patterns.

  Example: a pointing gesture may involve gaze, movement, facial expressions, and speech. Each of these modalities have different temporal pattens.

31

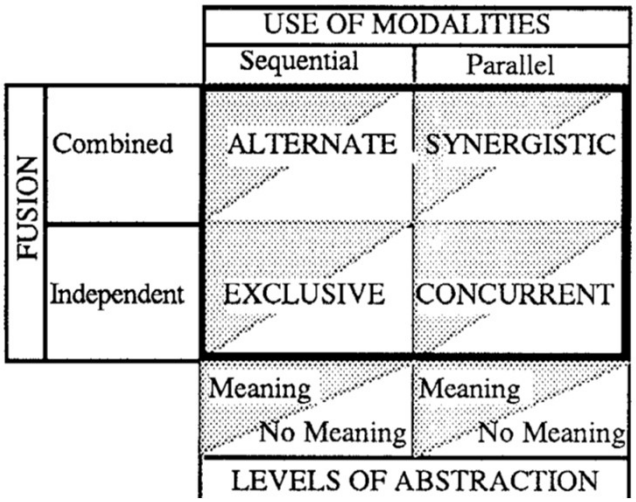
The CASE design space

- CASE (Concurrent, Alternate, Synergistic, or Exclusive) is a design space, i.e., a generic tool providing a common vocabulary for classifying and comparing systems designs, defined in (Nigay and Coutaz,1993).
- It is the first referenced model of multimodal combination.
- It defines three dimensions:
  - Level of abstraction: data is represented and processed at different levels of abstraction.
  - Use of modalities: temporal availability of modalities
  - Fusion: possible combination of different types of data

32

The CASE design space

- Following CASE, a multimodal system may belong to one of the eight subspaces of the design space.



Picture from (Nigay and Coutaz,1993)

33

The CASE design space

- CASE has several advantages:
  - The design space makes it explicit the way different modalities are supported by a particular system.
  - The classification scheme makes it precise the location of a system within the design space.
  - CASE is complete enough if the focus is solely on the machine side, that is, how a machine interprets a multimodal command.
  - It is the easiest model to understand due to its focus.
  - It gives developers a good idea of what kind of processing can be applied to multimodal commands.

Nigay, L., and Coutaz, J., 1993. A design space for multimodal systems: concurrent processing and data fusion. In Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems (CHI'93), 172-178.

34

The CARE properties

casa Paganini infomus

- CARE (Complementarity, Assignment, Redundancy, and Equivalence) is a way of characterizing and assessing aspects of a multimodal dialogue through a set of properties (Coutaz et al. 1995; Serrano and Nigay, 2009).
- The formal expressions of CARE relies on the notions of state, agent, goal, modality, and temporal relationship.

Coutaz, J., Nigay, L., Salber, D., Blandford, A., May, J., Young, R.M., 1995. Four Easy Pieces for Assessing the Usability of Multimodal Interaction: The Care Properties. In Nordby, K., Helmersen, P., Gilmore, D.J., Arnesen, S.A. (eds) Human—Computer Interaction. IFIP Advances in Information and Communication Technology.

Serrano, M., and Nigay. L., 2009. Temporal Aspects of CARE-based Multimodal Fusion: From a Fusion Mechanism to Composition Components and WoZ Components. In Proceedings of the 11th International Conference on Multimodal Interfaces (ICMI 2009).

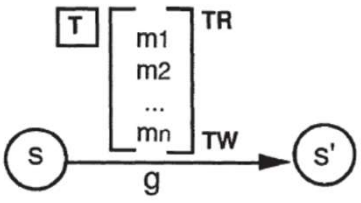
35

The CARE properties

casa Paganini infomus

- A state *s* is a set of properties measured at a particular time.
- An agent is an entity capable to perform actions.
- A goal *g* is a state that an agent intend to reach.
- A modality *m* is a method an agent can use to reach a goal.
- A temporal relationship TR characterizes the use of modalities over time (in a temporal window TW).

Notation for expressive CARE properties.
Picture from (Coutaz et al., 1995)

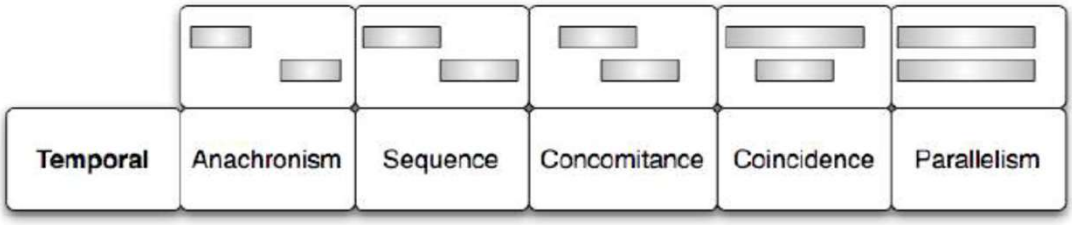


36

The CARE properties

casa Paganini infomus

- Temporality concerns temporal relationships between modalities, defined according to the Allen's interval logic.



Picture from (Serrano and Nigay, 2009)

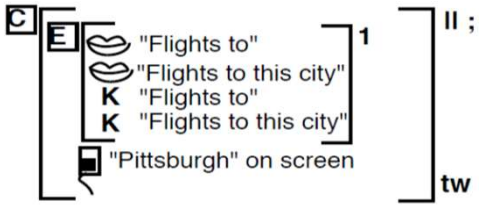
37

The CARE properties

casa Paganini infomus

- Complementarity: modalities are complementary to reach state *s'* from state *s* within a TW, if all of them must be used for reaching the target state, i.e., none of them taken individually can reach the target state.
- Assignment: a modality is assigned in state *s* to reach state *s'*, if no other modality can be used to reach *s'* from *s*. Assignment expresses the absence of choice.

Example of complementarity (use of speech, keyboard, and selection tool in a window). Picture from (Coutaz et al., 1995)

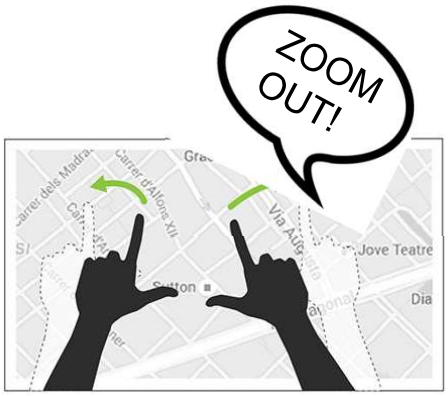


38

The CARE properties

casa Paganini infomus

- Redundancy: modalities are used redundantly to reach state *s'* from state *s*, if they have the same expressive power (i.e., they are equivalent) and if all of them are used in the same temporal window. In other words, the agent uses repetitive behavior without increasing its expressive power.



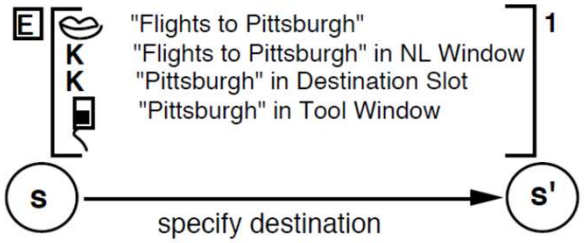
39

The CARE properties

casa Paganini infomus

- Equivalence: modalities are equivalent for reaching state *s'* from state *s*, if it is necessary and sufficient to use any one of them. Equivalence expresses the availability of a choice between multiple modalities, but it does not impose any form of temporal constraint on them.

Example of equivalence (use of speech, keyboard, and selection tool in a window). Picture from (Coutaz et al., 1995)



40

The CARE properties

casa Paganini infomus

- Some notes about CARE properties:
  - In contrast to redundancy, that does not favor any modality, complementarity can be driven by a dominant modality, that requires the use of others.
  - Equivalence and assignment both measure the choice available at some point in the multimodal dialogue.
  - Two different temporal relationships exist: sequentiality and parallelism having different implications on both usability and software development.

41

The U-CARE properties

casa Paganini infomus

- U-CARE properties are the counterpart of CARE properties on the user-side.
- User can have preferences due e.g., to knowledge, background, physical impairment and behavior, and so on.
- In a multimodal dialogue, CARE properties must match with the U-CARE properties, i.e., for each CARE property the supported modalities have to match (at least one!) with those preferred by the user.

U-CARE

- U-assignment
- U-equivalence
- U-redundancy
- U-complementarity

42

CARE and U-CARE compatibility

- Example 1
  U-assignment to Ua. For system Assignment to modality Sa, the condition for compatibility is that Sa = Ua. For system Equivalence or Redundancy over a set Se, the condition of compatibility is that Ua ∈ Se.
- Example 2
  U-equivalence where user can use any one of a set of modalities Ue. For system Assignment, the compatibility condition is that Sa ∈ Ue. For system Equivalence the condition is that Ue ∩ Se ≠ ∅.

43

Frameworks

- Frameworks and conceptual models have been proposed for multimodal systems.
- Frameworks are not architectures, rather they are a level of abstraction above an architecture.
- I.e., they show components and their connections, but they do not include implementation details (e.g., how components are allocated to hardware devices).
- Frameworks have been developed for both multimodal systems focusing on verbal communication and for multimodal systems focusing on non-verbal communication.

44

The W3C Framework

casa Paganini infomus

- Identifies the major components of a multimodal system. Each represents a set of related functions.
- Identifies the markup languages to describe information required by components and data flowing.

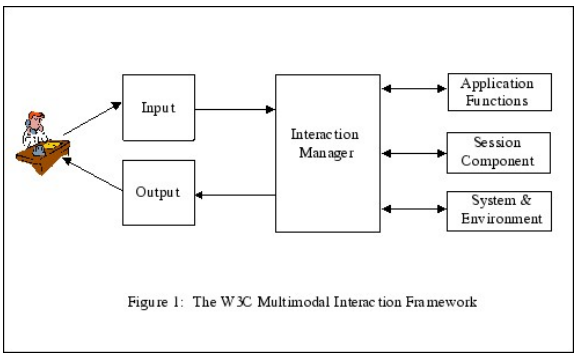


Figure 1: The W3C Multimodal Interaction Framework

W3C Multimodal Interaction Working Group, Multimodal Interaction Requirements, W3C NOTE 6 May 2003, https://www.w3.org/TR/mmi-framework/

45

The W3C Framework

casa Paganini infomus

- Input component

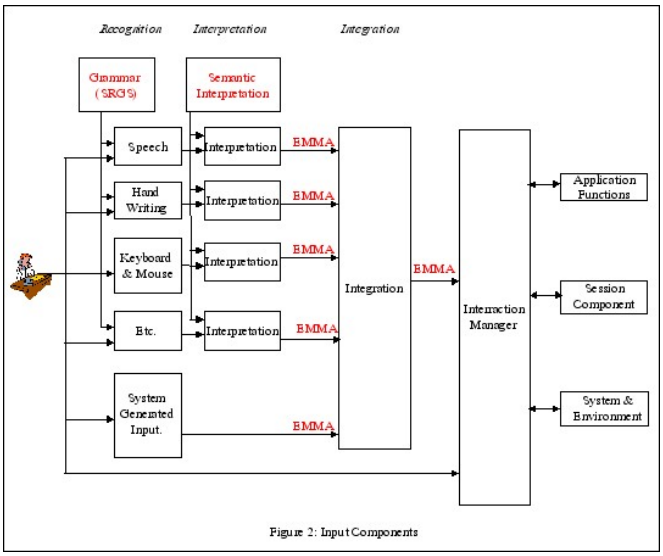


Figure 2: Input Components

46

The W3C framework

casa Paganini infomus

- EMMA – Extensible MultiModal Annotation markup language.

```
<emma:emma version="1.0"
    xmlns:emma="http://www.w3.org/2003/04/emma"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.w3.org/2003/04/emma
     http://www.w3.org/TR/2009/REC-emma-20090210/emma.xsd"
    xmlns="http://www.example.com/example">
  <emma:one-of id="r1" emma:start="1087995961542" emma:end="1087995963542"
     emma:medium="acoustic" emma:mode="voice">
    <emma:interpretation id="int1" emma:confidence="0.75"
    emma:tokens="flights from boston to denver">
      <origin>Boston</origin>
      <destination>Denver</destination>
    </emma:interpretation>

    <emma:interpretation id="int2" emma:confidence="0.68"
    emma:tokens="flights from austin to denver">
      <origin>Austin</origin>
      <destination>Denver</destination>
    </emma:interpretation>
  </emma:one-of>
</emma:emma>
```

47

The W3C Framework

casa Paganini infomus

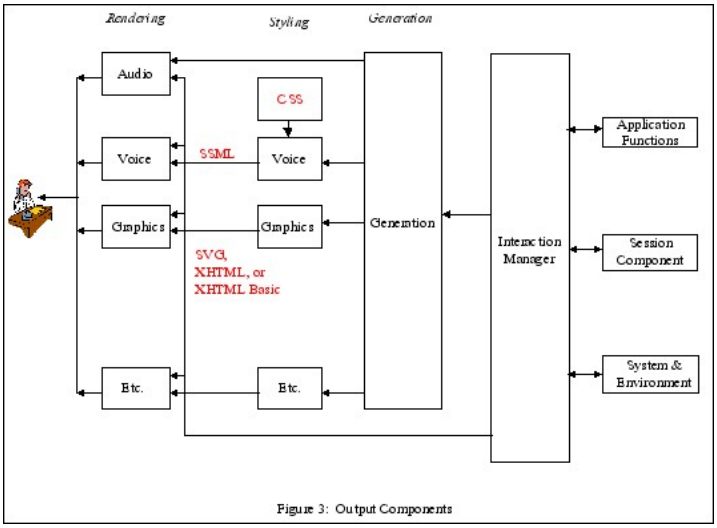


Figure 3: Output Components

- Output component

48

The W3C Framework

casa Paganini infomus

- Interaction manager:
  - Coordinates data and manages execution flow.
  - Maintains the interaction state and context of the application and responds to inputs and changes in the system and environment.
  - Manages these changes and coordinates input and output across component interface objects.
  - In some architectures is one single component; in others is a composition of (possibly distributed) components.
- Application functions: manages interaction with the applications that use the framework.

49

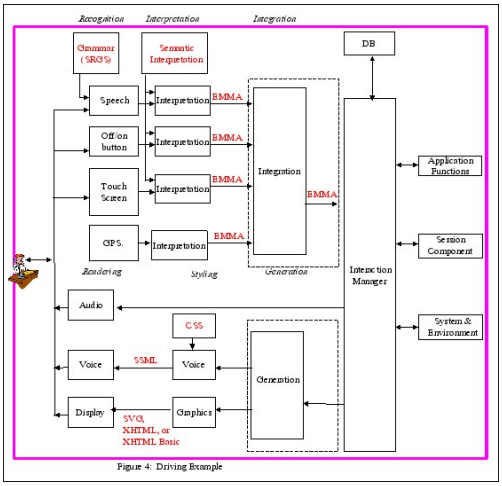
The W3C Framework

casa Paganini infomus

- Session component: provides an interface to the interaction manager to support state management, and temporary and persistent sessions. This is useful e.g., when an application runs on multiple devices; an application is session based e.g., multiplayer game; an application provides multiple input/output modes.
- System and Environment component: enables the interaction manager to find out about and respond to changes in device capabilities, user preferences, and environmental conditions. For example, which of the available modes, the user wishes to use — has the user muted audio input?

50

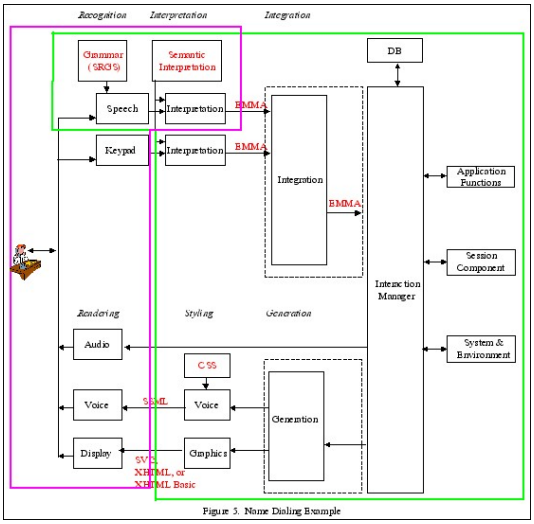
The W3C Framework



Example1: while driving the user wants to take a detour to a local restaurant.
She initiates service via a button on her steering wheel and interacts with the system via the touch screen and speech.

51

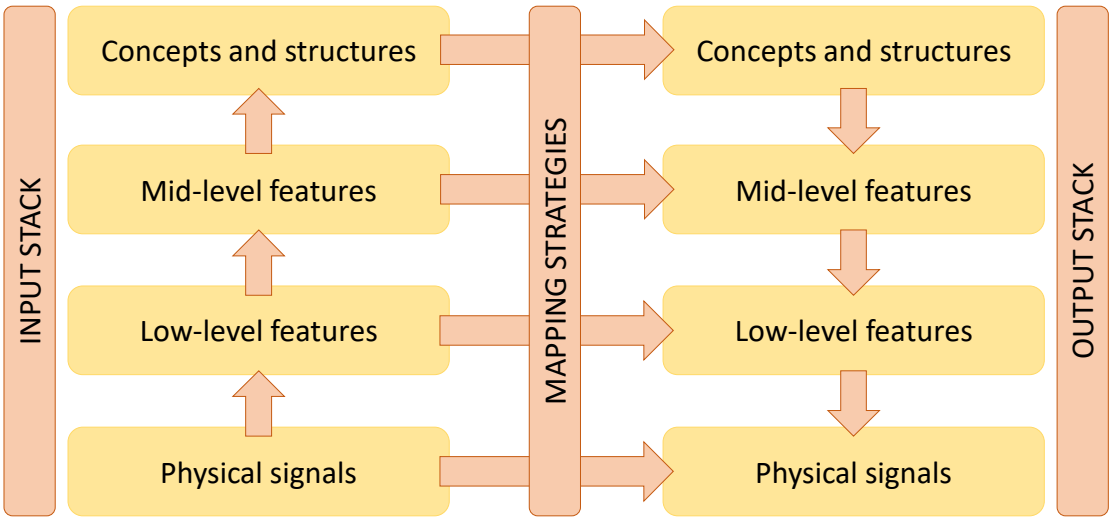
The W3C Framework



Example 2: a user initiates a phone call by saying the name of the person to be contacted. Visual and spoken dialogs are used to narrow selection, and to exchange multimedia messages if the called person is unavailable. Call handling is done by a script provided by the called person.

52

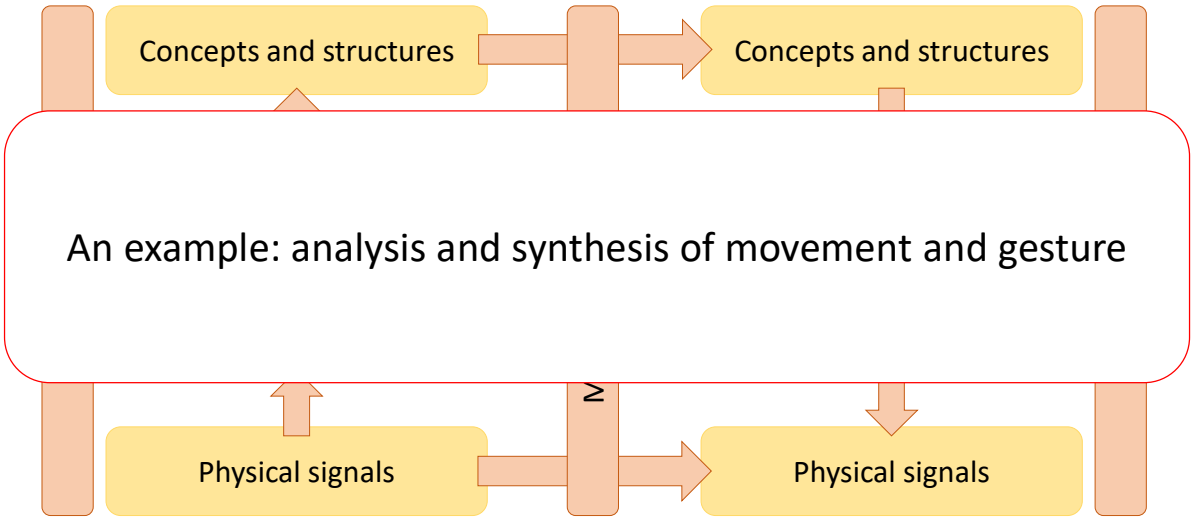
A layered framework



Camurri, A., Mazzarino, B., Ricchetti, M., Timmers, R., and Volpe, G., 2004. Multimodal analysis of expressive gesture in music and dance performances. In A. Camurri, G. Volpe (Eds.), Gesture-based Communication in Human-Computer Interaction, LNAI 2915, 20-39.

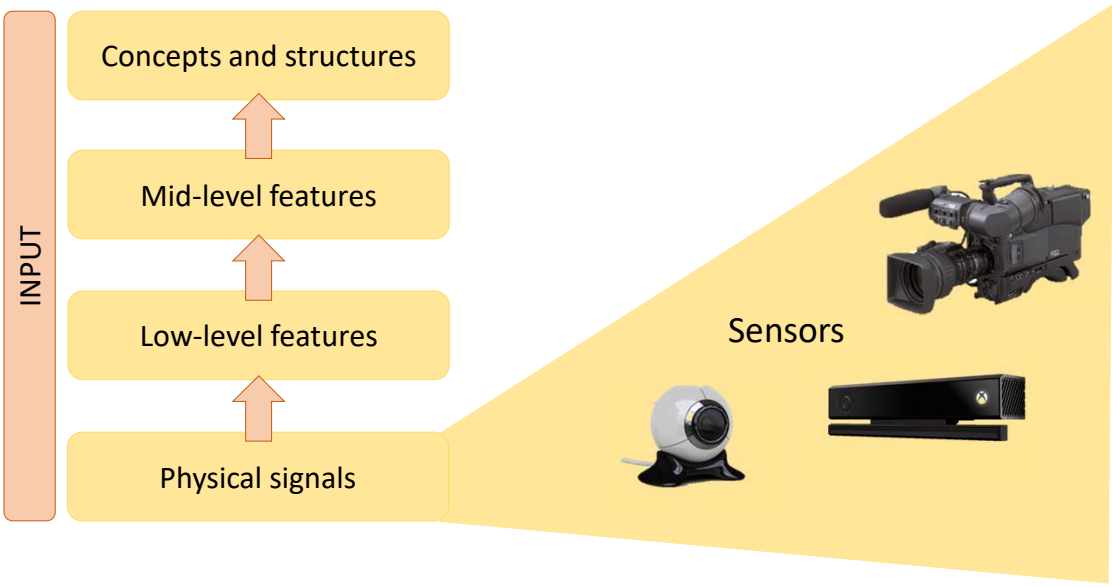
53

A layered framework



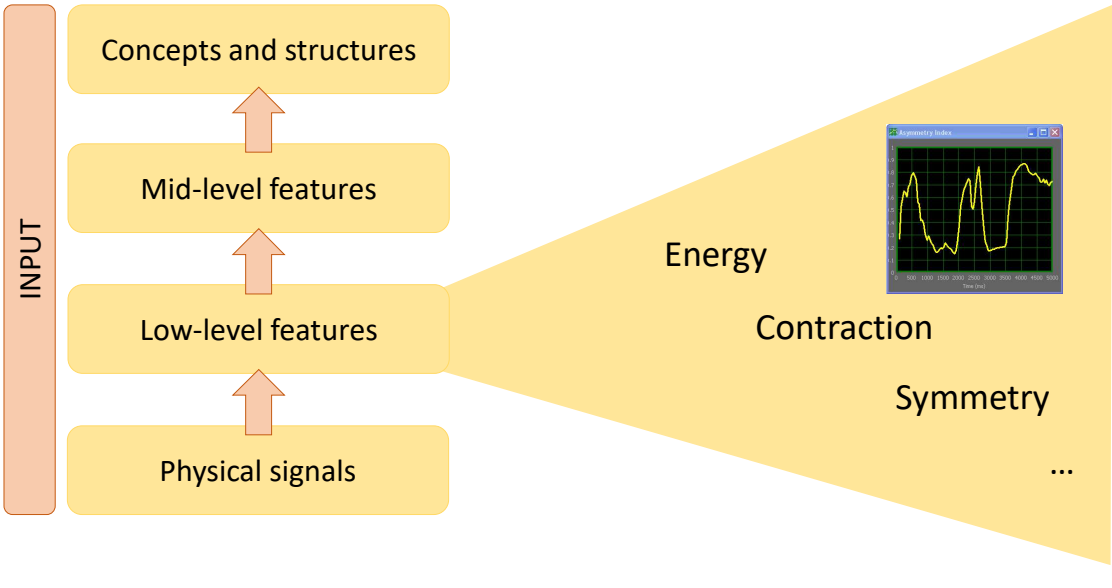
54

A layered framework



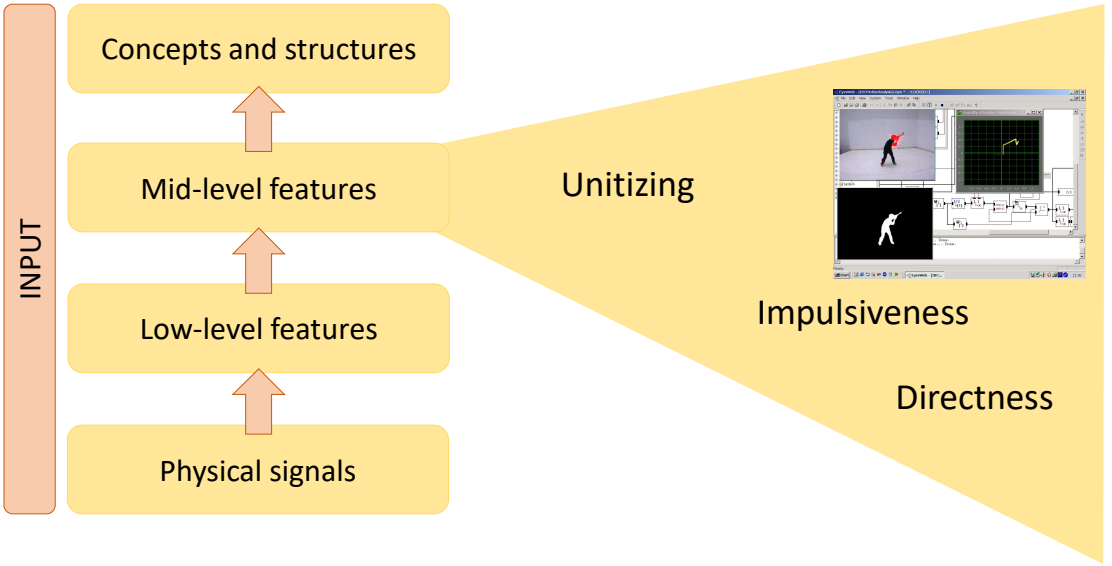
55

A layered framework



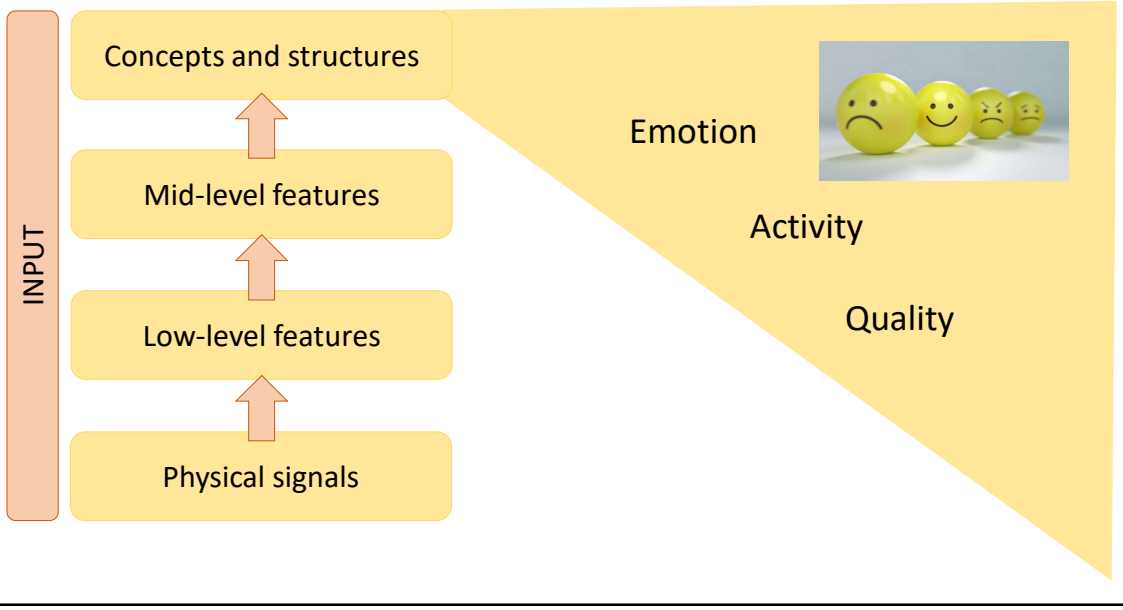
56

A layered framework



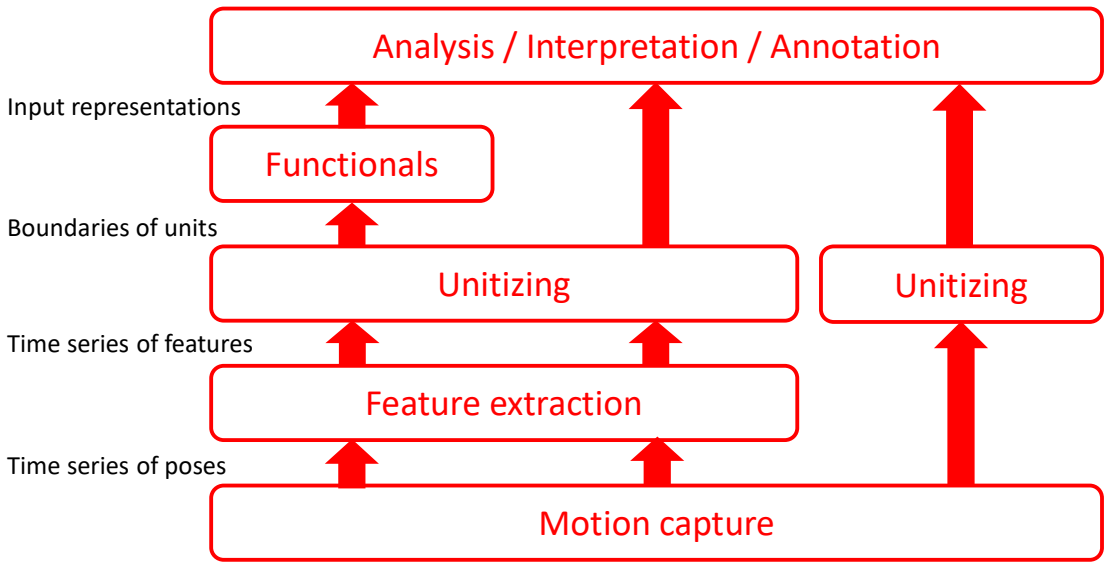
57

A layered framework



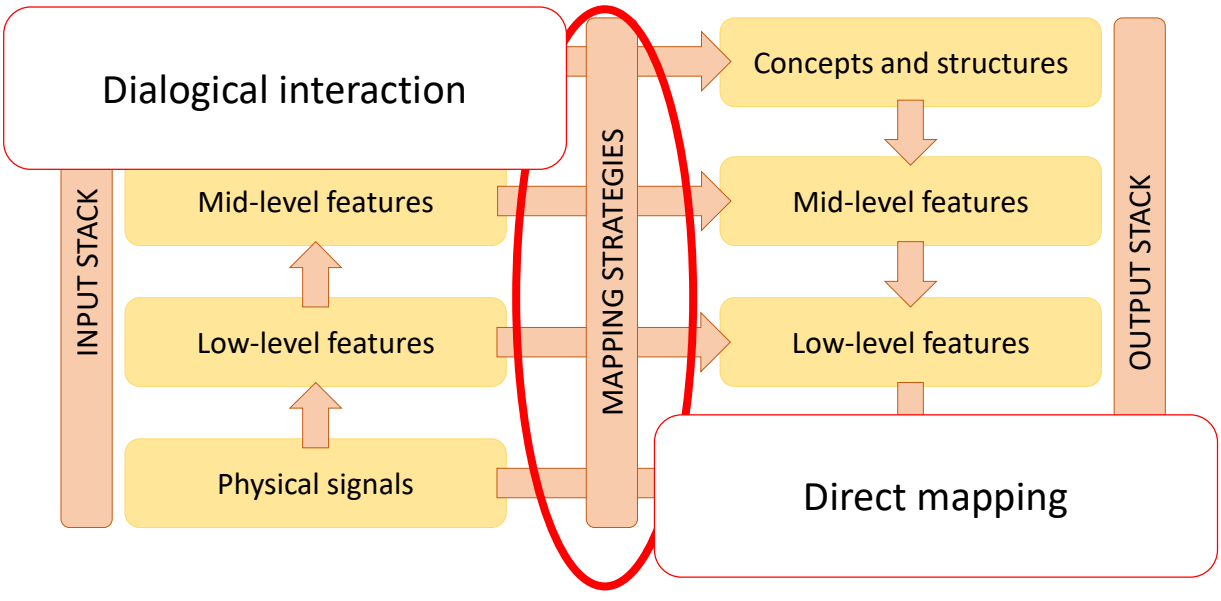
58

An updated input pipeline



59

A layered framework



60

A layered framework

Emotion

Activity

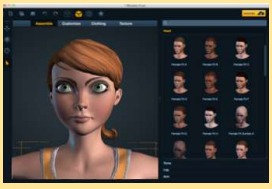
Quality

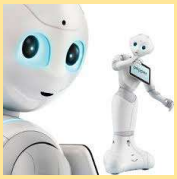
- Concepts and structures
- Mid-level features
- Low-level features
- Physical signals

OUTPUT

61

A layered framework





Choice of gestures

- Concepts and structures
- Mid-level features
- Low-level features
- Physical signals

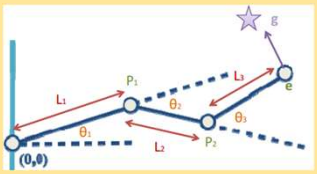
OUTPUT

62

A layered framework

casa Paganini infomus

Synthesis of movement of parts of the body



- Concepts and structures
- Mid-level features
- Low-level features
- Physical signals

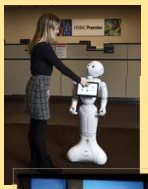
OUTPUT

63

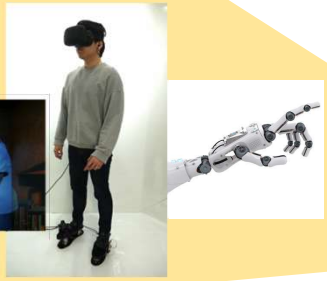
A layered framework

casa Paganini infomus

Rendering



Actuators

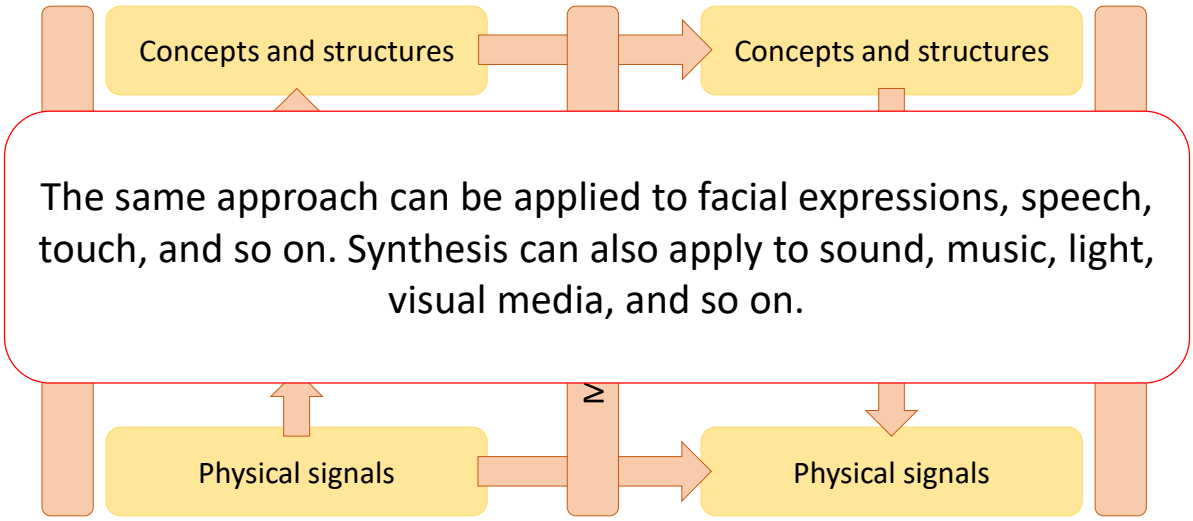


- Concepts and structures
- Mid-level features
- Low-level features
- Physical signals

OUTPUT

64

A layered framework



65

Ten Myths (Oviatt, 1999)

- Is everything ok?
- I.e., if we apply guidelines, models, and frameworks can we always succeed in designing an effective, efficient, and satisfactory multimodal system?

Source: https://www.nettl.com/uk/5-user-experience-uiux-ecommerce-tips/

66

Ten Myths (Oviatt, 1999)

casa Paganini infomus

1. If you build a multimodal system, users will interact multimodally.
   Users tend to intermix unimodal and multimodal interactions.

2. Speech and pointing is the dominant multimodal integration pattern.
   Modalities that transmit written input, manual gesturing, and facial expressions can generate symbolic information that is more richly expressive than simple object selection.

3. Multimodal input involves simultaneous signals.
   Multimodal signals often do not co-occur temporally; much of multimodal interaction involve sequential use of modalities.

67

Ten Myths (Oviatt, 1999)

casa Paganini infomus

4. Speech is the primary input mode in any multimodal system that includes it.
   Speech is neither the exclusive carrier of important content, nor does it have temporal precedence over other modalities. These can convey information that is not present in the speech signal, e.g., spatial information and manner of action.

5. Multimodal language does not differ linguistically from unimodal language.
   Multimodal language is briefer, syntactically simpler, and less disfluent than users' unimodal speech.

68

Ten Myths (Oviatt, 1999)

casa Paganini infomus

6. Multimodal integration involves redundancy of content between modes.
   Complementarity of content may be more significant in multimodal systems than redundancy.

7. Individual error-prone recognition technologies combine multimodally to produce even greater unreliability.
   In an appropriately flexible multimodal interface, people determine how to use the available input modalities most effectively; mutual disambiguation of signals may contribute to a higher level of robustness.

69

Ten Myths (Oviatt, 1999)

casa Paganini infomus

8. All users' multimodal commands are integrated in a uniform way.
   When users interact multimodally, there actually can be individual differences in integration patterns. Systems should adapt to a user's dominant integration pattern.

9. Different input modes are capable of transmitting comparable content.
   Modalities differ in the type of information they transmit, their functionality during communication, the way they are integrated with other modes, and in their basic suitability to be incorporated into different interface styles.

70

Ten Myths (Oviatt, 1999)

casa Paganini infomus

10. Enhanced efficiency is the main advantage of multimodal systems.
    Their main advantages may be found in other aspects, such as decreased errors, increased flexibility, or increased user satisfaction.

- The design of multimodal systems that blend input modes synergistically depends on intimate knowledge of the properties of different modalities and the information content they carry, what characteristics are unique to multimodal language and its processability, and how multimodal input is integrated and synchronized.

Oviatt., S., 1999. Ten myths of multimodal interaction. Communications of the ACM 42, 11, 74-81.

71

Casa Paganini – InfoMus
DIBRIS – University of Genoa
www.casapaganini.org

Multimodal Systems

Gualtiero Volpe
gualtiero.volpe@unige.it

1

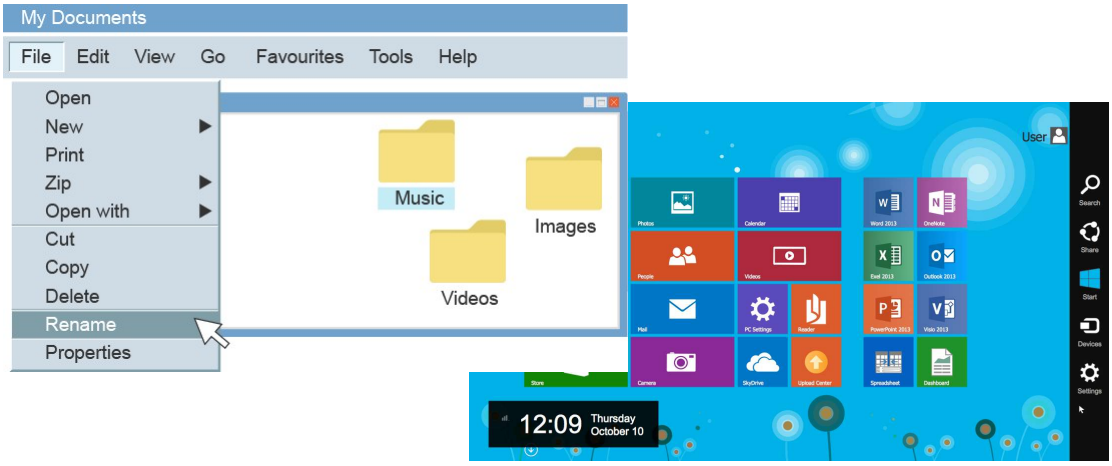
1. Introduction to Multimodal Systems

2

WIMP

casa Paganini infomus

- Most traditional, consolidate, commonly used, and widespread interfaces are graphical user interfaces (GUIs).



3

WIMP

casa Paganini infomus

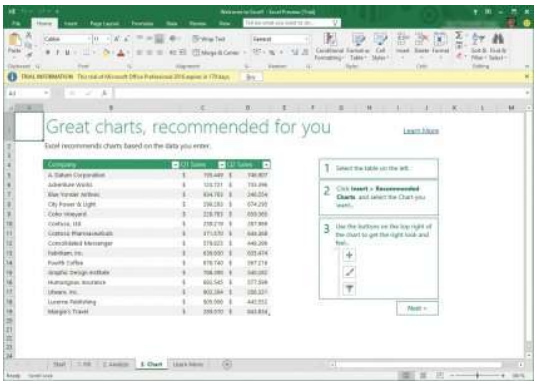
- WIMP interfaces adopt the so-called WIMP paradigm, i.e., Windows, Icons, Menus and a Pointing device.

4

WIMP

casa Paganini infomus

- WIMP interfaces have become so prevalent since they are very good at abstracting workspaces, documents, and their actions.
- Their analogous paradigm to documents as paper sheets or folders makes WIMP interfaces easy to introduce to novice users.

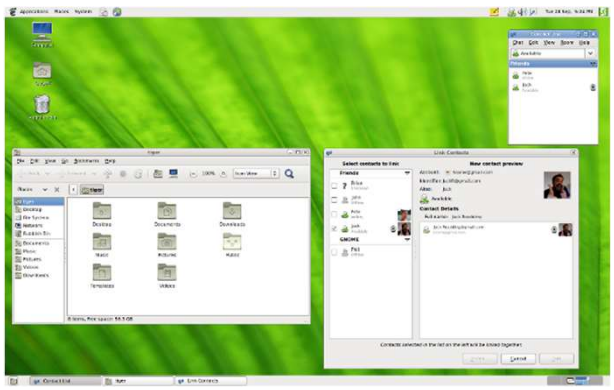


5

WIMP

casa Paganini infomus

- Furthermore, their basic representations as rectangular regions on a 2D flat screen make them a good fit for system programmers.
- Also, generality makes them very suitable for multitasking work environments.



6

The desktop metaphor

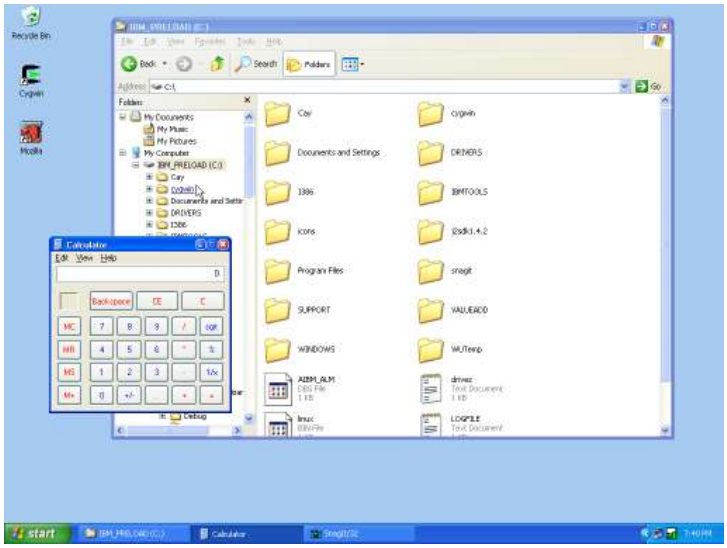
casa Paganini infomus

- The WIMP paradigm is commonly exploited to implement the broadly used desktop metaphor, a unifying concept for GUIs.
- It was first introduced by Alan Kay at Xerox PARC in 1970.
- The first computer to popularize it, using it as a standard feature over the earlier command-line interface was the Apple Macintosh in 1984.
- The desktop metaphor treats the monitor of a computer as the user's desktop, upon which objects, such as documents and folders of documents can be placed.
- A document can be opened into a window, which represents a paper copy of the document placed on the desktop.
- Small applications called desk accessories are available, e.g., desk calculator, notepad, and so on.

7

The desktop metaphor

casa Paganini infomus



Source: http://www.cs.sjsu.edu/web_mater/cs46a/cs46alab/lab1/tutorial.html

8

Post-WIMP

casa Paganini infomus

- WIMP interfaces are not optimal for complex tasks such as computer-aided design or for applications needing more natural interaction paradigms, e.g., interactive games.
- Post-WIMP interfaces (Van Dam, 1997) aim at overcoming such problems and consist of widgetless user interfaces, including virtual reality systems, and user interfaces based on gestures, speech, and physical controls.
- Post-WIMP interfaces integrate input from several sensory channels and produce multimedia output.

van Dam, A., 1997. Post-WIMP user interfaces. Communications of the ACM 40, 2, 63-67.

9

Post-WIMP: automotive

casa Paganini infomus

- Affectiva Interior Sensing AI.

Source: https://go.affectiva.com/auto; https://youtu.be/4duUxSFV9qU

10

Post-WIMP: sport

casa Paganini infomus

- Vicon Biomechanics and Sports Science Showreel

Source: https://www.youtube.com/watch?v=uPn26JbRN4g

11

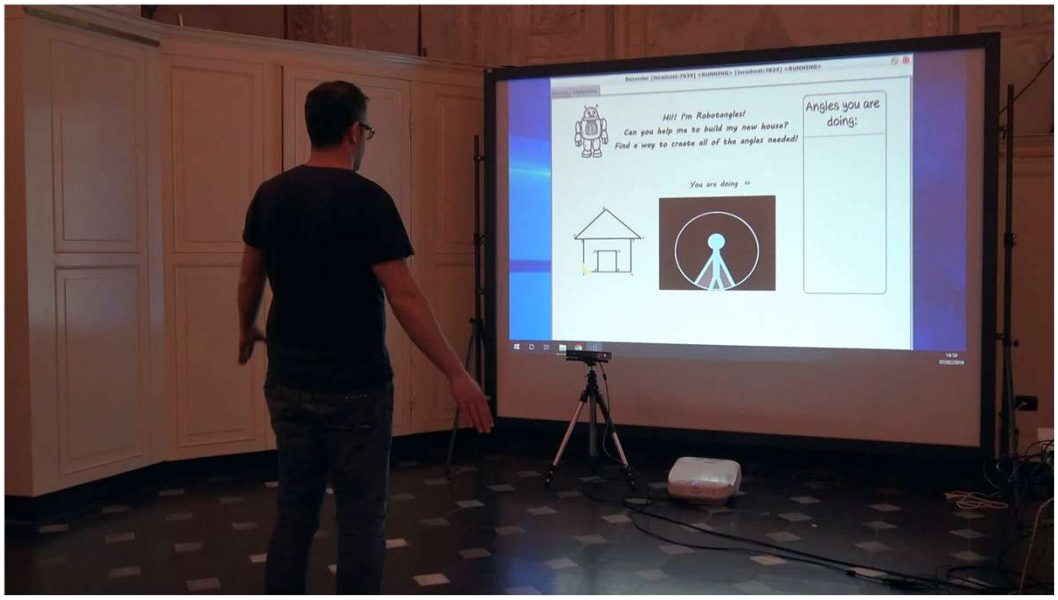
Post-WIMP: performing arts

casa Paganini infomus

12

Post-WIMP: education

casa Paganini infomus



13

Post-WIMP: rehabilitation

casa Paganini infomus

14

A simplified processing pipeline

casa Paganini infomus



Devices



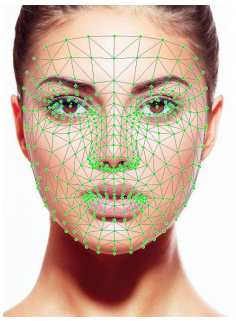
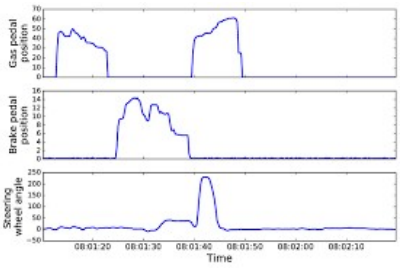


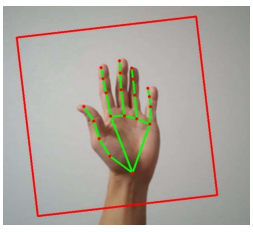
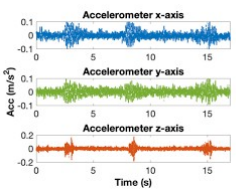

15

A simplified processing pipeline

casa Paganini infomus

Input Devices → Raw Data

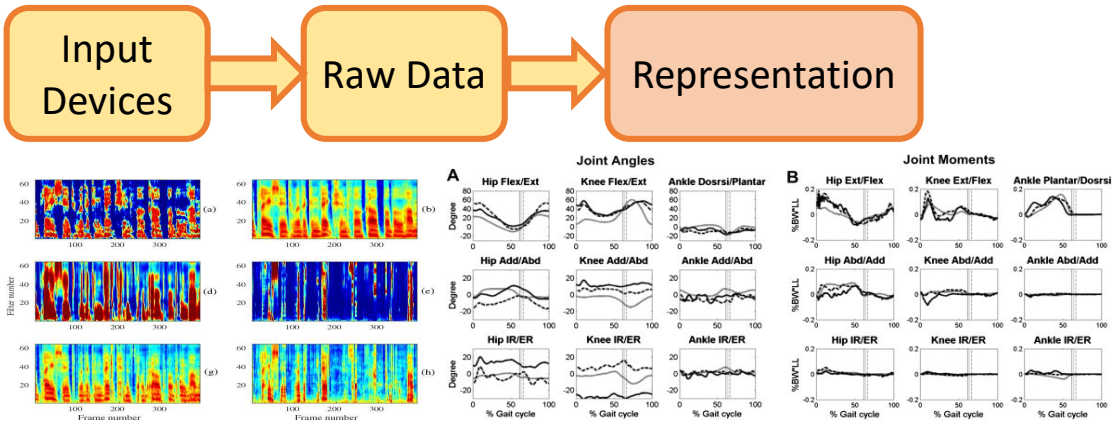




16

A simplified processing pipeline

casa Paganini infomus

A collection of features (either manually identified or learned) that properly describe the target phenomenon.

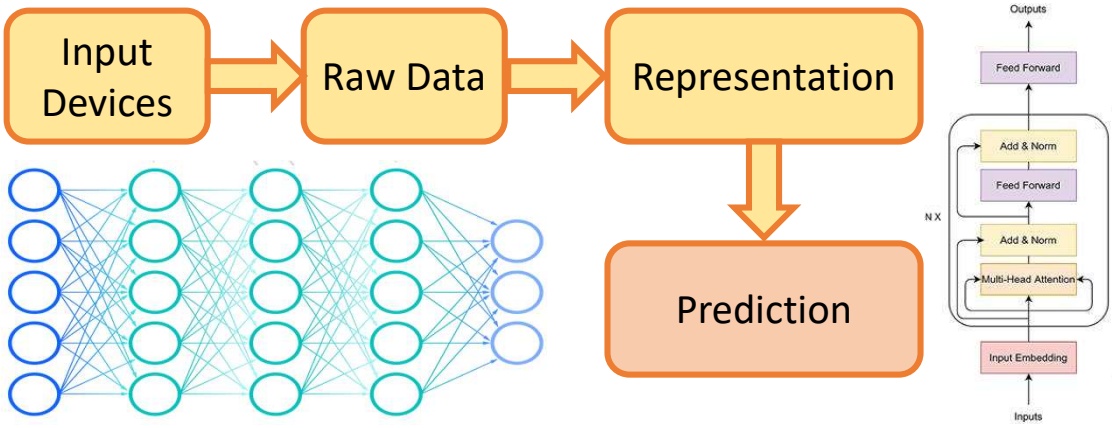


17

A simplified processing pipeline

casa Paganini infomus

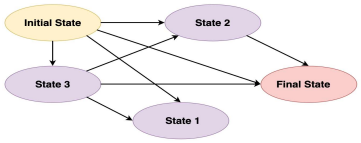
Machine learning is usually applied.

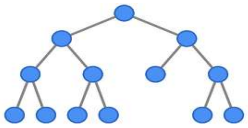


18

A simplified processing pipeline

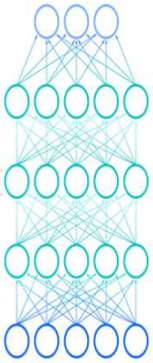
casa Paganini infomus

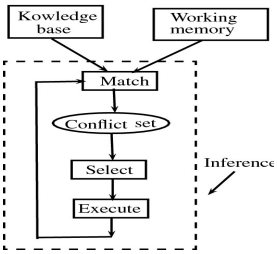




Rule-based systems, trees, planners, machine learning …

Input Devices → Raw Data → Representation → Prediction → Mapping

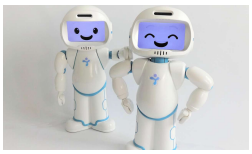




19

A simplified processing pipeline

casa Paganini infomus

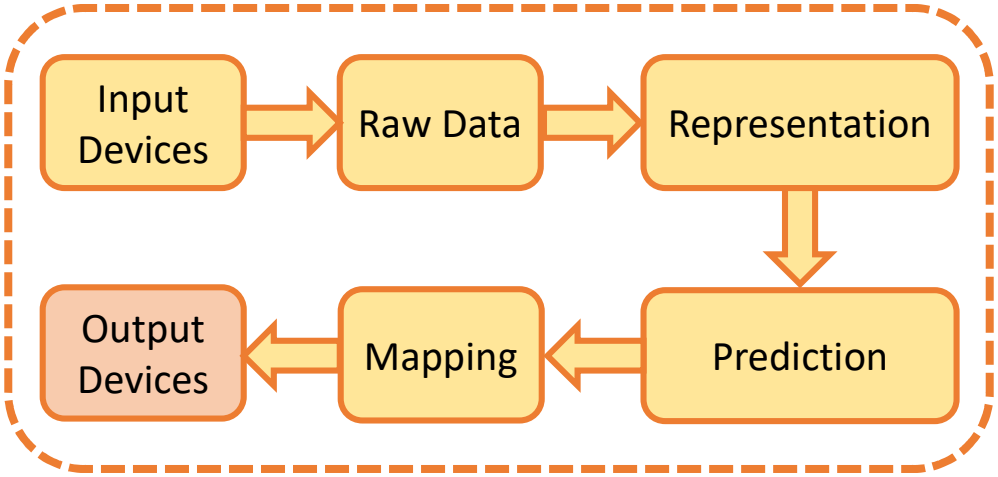


Input Devices → Raw Data → Representation → Prediction → Mapping → Output Devices

20

A simplified processing pipeline

When multiple sensory modalities are involved, this is understood as a multimodal system.



21

Sensory modalities

- Sensory modality: the sensory channel through which information is perceived; it refers to the communication channel used for transferring or acquiring information.
- Multimodal therefore refers to the integration of information from several different sensory channels.

22

Multimodal systems

casa Paganini infomus

- Multimodal systems are "systems that support a user communicating with an application by using different modalities such as voice (in a human language), gesture, handwriting, typing, audio-visual speech, etc." (W3C Multimodal Interaction Working Group, 2008)
- "A multimodal HCI system is simply one that responds to inputs in more than one modality or communication channel (e.g., speech, gesture, writing, and others)." (Jaimes and Sebe, 2007)

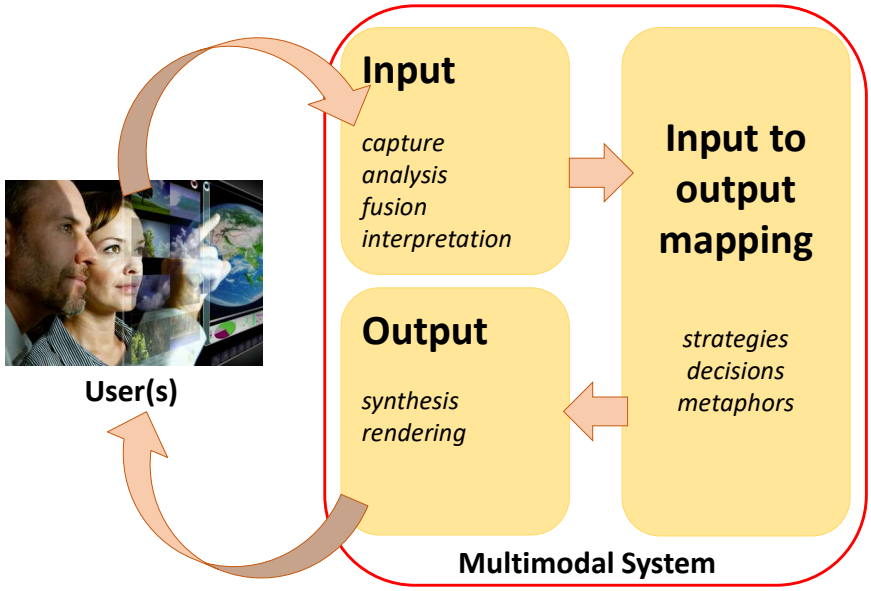
W3C Multimodal Interaction Working Group, Multimodal Interaction Requirements, W3C NOTE 8 January 2003, http://www.w3.org/TR/mmi-reqs/

Jaimes, A., and Sebe, N., 2007. Multimodal human–computer interaction: A survey. Computer Vision and Image Understanding, 108, 116–134.

23

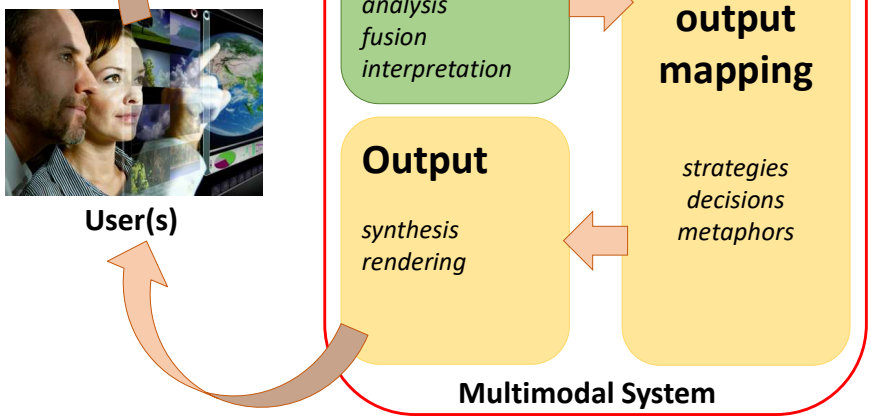
Multimodal systems

casa Paganini infomus



24

Multimodal systems: input



25

Multimodal systems: input

- The system observes the user's behavior.
- Input from several modalities is captured by sensors.
- Captured input streams are typically subject to processing:
  - Analysis: features characterizing the input stream (i.e., the user's behavior) for each modality are automatically extracted.
  - Fusion: features from the single modalities are integrated.
- Results are interpreted to build semantic representations, e.g., in terms communicated meaning, cognitive and/or emotional states (goals, beliefs, mood, emotion, and so on).
- For instance, speech input may be input to a speech recognition engine, e.g., including semantic interpretation.

26

Multimodal systems: mapping

User(s)

Input

capture
analysis
fusion
interpretation

Output

synthesis
rendering

Input to output mapping

strategies
decisions
metaphors

Multimodal System

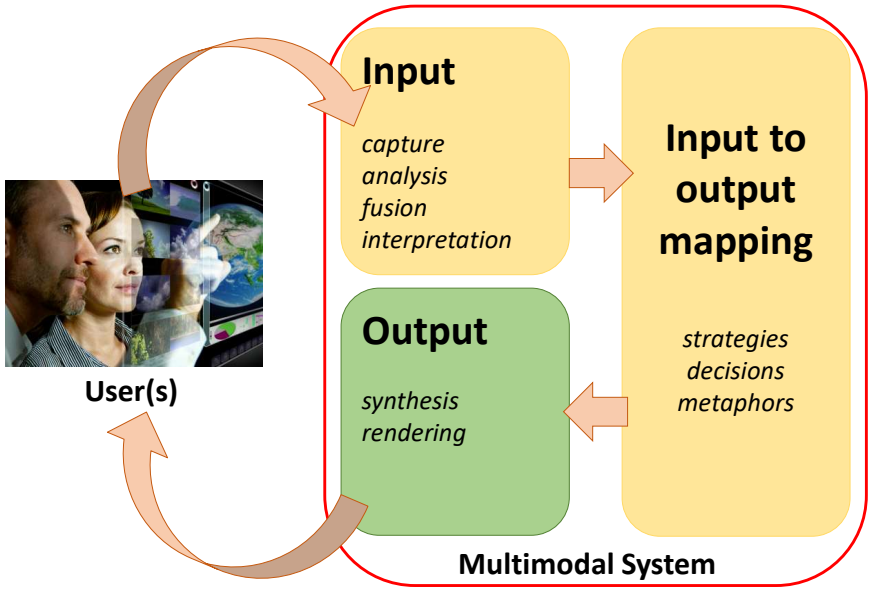
27

Multimodal systems

- The synchronization behavior of a multimodal system describes the way in which any input in one modality is reflected into the output in another modality, as well as the way input is combined across modalities (coordination capability). Implementing it may encompass:
  - Strategies: rules / procedures to map input into output.
  - Decisions: made according to the selected strategies and the input the system actually received (i.e., strategies are applied).
  - Metaphors: ways of relating computing (i.e., usage of a multimodal system) to other common real-world activities.

28

Multimodal systems: output



29

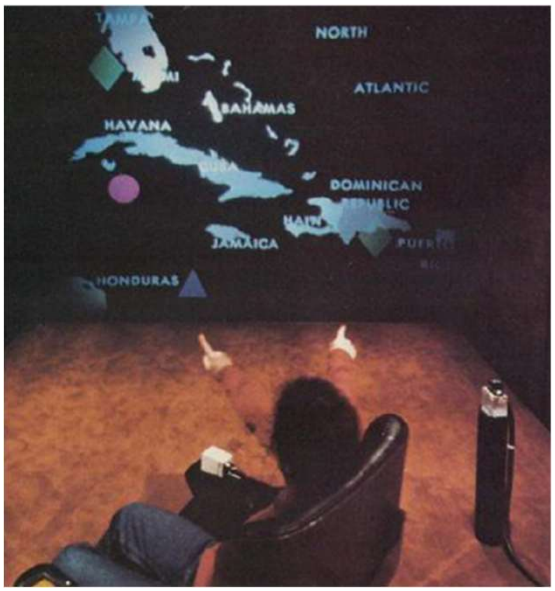
Multimodal systems: output

- The output generated by a multimodal system consists of real-time multimedia feedback for the user(s), based on analysis of the input, internal models, and tasks at hand.
- It can take various forms, e.g., audio, visual, and haptic feedback, lighting, and so on.
- The output generation process consists of:
  - Synthesis: models of the output are generated describing how the output will look like, once rendered.
  - Rendering: computer programs and devices are used to actually produce the output.

30

The first multimodal system

casa Paganini infomus



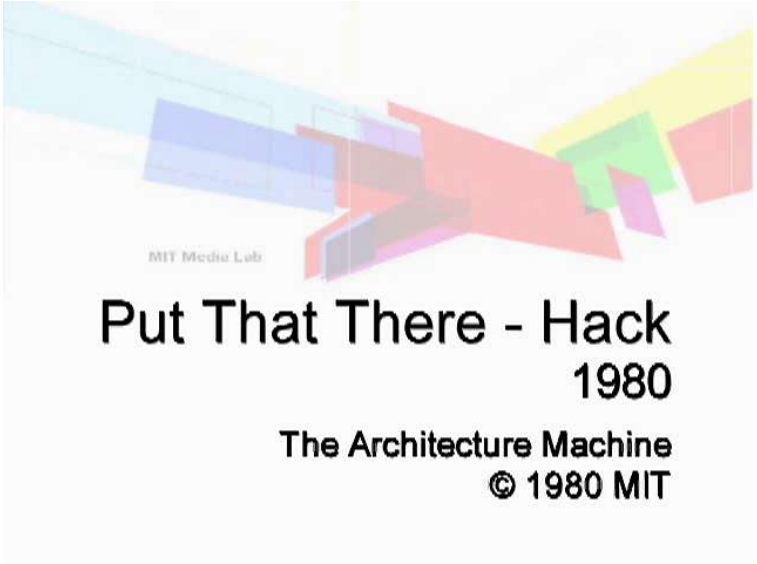
Put That There
(Bolt, 1980)

Bolt, R.A., 1980. "Put-that-there": Voice and gesture at the graphics interface. In Proceedings of the 7th annual conference on Computer graphics and interactive techniques (SIGGRAPH'80), 262-270.

31

The first multimodal system

casa Paganini infomus



32