FUN ROBOTIC
OUTDOOR GUIDE

Deliverable: D3.2

Demonstrator of human affective signals analyzer

Consortium

UNIVERSITEIT VAN AMSTERDAM (UvA)
YDREAMS - INFORMATICA S.A. (YD)
IDMIND - ENGENHARIA DE SISTEMAS LDA (IDM)
UNIVERSIDAD PABLO DE OLAVIDE (UPO)
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE (ICL)
UNIVERSITY OF TWENTE (UT)

Grant agreement no. **288235**
Funding scheme **STREP**

UNIVERSITEIT TWENTE.

DOCUMENT INFORMATION

Project	
Project acronym:	FROG
Project full title:	Fun Robotic Outdoor Guide
Grant agreement no.:	288235
Funding scheme:	STREP
Project start date:	1 October 2011
Project duration:	36 Months
Call topic:	ICT-2011.2.1 Cognitive Systems and Robotics (a), (d)
Project web-site:	www.frogrobot.eu

Document	
Deliverable number:	D3.2
Deliverable title:	Demonstrator of human affective signals analyzer
Due date of deliverable:	M30 – 31 March 2014
Actual submission date:	M32 – 28 May 2014
Editors:	Imperial
Authors:	Imperial
Reviewers:	UT
Participating beneficiaries:	Imperial
Work Package no.:	3
Work Package title:	Behaviour detection and human-aware guidance
Work Package leader:	Imperial
Work Package participants:	1,4,5
Estimated person-months for deliverable:	10
Dissemination level:	Public
Nature:	Demonstrator
Version:	
Draft/Final:	Final
No of pages (including cover):	33
Keywords:	Affective signal analysis, Interest level prediction, Annotation alignment

Chapter 1

Introduction

The main concept of FROG is to deliver a robust autonomous mobile robot that uses innovative design and behavior to engage visitors in the exploration of outdoor sites. The robot's human-aware behaviors and interaction will be developed with the specific measurable goals to enable user engagement and interest in the resources the site has to offer, knowledge transfer, ease of use and enjoyability.

FROG uses tour guide strategies derived from contextual observation studies in order to engage the visitor in learning more about the points of interests they encounter. FROG will adapt its tour guide strategy behaviors based on the visitors' implicit affective feedback (e.g. attention and interest). Also, FROG is planned to know whether visitors are still moving, following or have stopped and whether they are interested and are paying attention. By detecting head pose and viewing direction FROG also knows what point of interest visitors are focused on.

The task in this work package aims further to develop a set of visual methods for detecting human affective states including users' positive and negative reactions to FROG robot and their overall level of interest and engagement in the current interaction with FROG. Detection is based on the state of the art in cognitive sciences and based on morphological and temporal correlations. For this purpose, facial landmarks as well as face pose were used. The method for facial landmark detection as well as for face pose estimation was developed for FROG and it was presented in work package 3.1 (FROG tracker). Firstly, interest annotations were obtained by annotated thousands of images based on FROG data. Secondly, temporal alignment and fusion of multiple annotations in time was performed. Finally, the annotations were used for training models for interest prediction. These tree main steps are presented in detail in the next chapters.

Chapter 2

Obtaining Interest Annotations

The interest annotations obtained in a continuous scale where obtained by providing the annotators with the following instructions:

- *Interest Rating in* $[-1, -0.5)$: the subject is *disinterested* in the interaction, can be mostly passive or appear bored, does not follow the robot and possibly wants to stop the session.
- *Interest Rating in* $[-0.5, 0)$: the subject appears passive, possibly hesitating to respond. The subject appears *indifferent* and *unmotivated*.
- *Interest Rating approx.* 0: the subject seems to follow the interaction with the interaction partner, but it can not be recognized if he/she is interested. The subject is *neutral*.
- *Interest Rating in* $(0, 0.5]$: The subject seems eager follow the interaction. The subject is *interested*.
- *Interest Rating in* $(0.5, 1]$: The subject seems pleased to participate in the interaction, can show some signs of *enthusiasm*, is expressive in terms of (positive) emotions (e.g., laughing,).

The interest annotations where quantised as follows:

- No Interest (Class 0) *Interest Rating in* $[-1, 0)$: *Disinterest* or *indifference*, the subject is not interested in the interaction and is unmotivated to participate, possibly wants to terminate the interaction or is neutral.

- Interested (Class 1) *Interest Rating in* $[0, 0.5)$: The subject seems interested in the interaction and appears eager to follow.

- Highly Interested (Class 2) *Interest Rating in* $[0.5, 1]$: The subject appears pleased to participate in the interaction, can show signs of enthusiasm and is expressive in terms of positive emotions (e.g., laughing).

Chapter 3

Analysis and Fusion of Continuous Sets of Annotations

Fusing multiple continuous expert annotations is a crucial problem in machine learning and computer vision, particularly when dealing with uncertain and subjective tasks related to affective behaviour. Inspired by the concept of inferring shared and individual latent spaces in Probabilistic Canonical Correlation Analysis (PCCA), we used a novel, generative model that discovers temporal dependencies on the shared/individual spaces (Dynamic Probabilistic CCA, DPCCA). In order to accommodate for temporal lags, which are prominent amongst continuous annotations, we further introduce a latent warping process, leading to the DPCCA with Time Warpings (DPCTW) model. Finally, we used two supervised variants of DPCCA/DPCTW which incorporate inputs (i.e. visual or audio features), both in a generative (SG-DPCCA) and discriminative manner (SD-DPCCA). We show that the resulting family of models (i) can be used as a unifying framework for solving the problems of temporal alignment and fusion of multiple annotations in time, (ii) can automatically rank and filter annotations based on latent posteriors or other model statistics, and (iii) that by incorporating dynamics, modelling annotation-specific biases, noise estimation, time warping and supervision, DPCTW outperforms state-of-the-art methods for both the aggregation of multiple, yet imperfect expert annotations as well as the alignment of affective behaviour.

We initially present the first generalisation of PCCA to learning temporal dependencies in the shared/individual spaces (Dynamic PCCA, DPCCA). By further augmenting DPCCA with time warping, the resulting model (Dynamic PCCA with Time Warpings, DPCTW) can be seen as a unifying framework, concisely applied to both problems. The individual contributions of this work can be summarised as follows:

- In comparison to state-of-the-art approaches in both fusion of multiple annotations and sequence alignment, our model bears several advantages. We assume that the "true" annotation/sequence lies in a shared latent space. E.g., in the problem of fusing multiple emotion annotations, we know that the experts have a common training in annotation. Nevertheless, each carries a set of individual factors

which can be assumed to be uninteresting (e.g., annotator/sequence specific bias). In our model, individual factors are accounted for within an annotator-specific latent space, thus effectively preventing the contamination of the shared space by individual factors. Most importantly, we introduce latent-space dynamics which model temporal dependencies in both common and individual signals. Furthermore, due to the probabilistic and dynamic nature of the model, each annotator/sequence's uncertainty can be estimated for each *sample*, rather than for each sequence.

- In contrast to current work on fusing multiple annotations, we use a novel framework able to handle temporal tasks. In addition to introducing dynamics, we also employ temporal alignment in order to eliminate temporal discrepancies amongst the annotations.

- We present an elegant extension of DTW-based sequence alignment techniques (e.g., Canonical Time Warping, CTW) to a probabilistic multiple-sequence setting. We accomplish this by treating the problem in a generative probabilistic setting, both in the static (multiset PCCA) and dynamic case (Dynamic PCCA).

3.1 Multiset Probabilistic CCA

We consider the probabilistic interpretation of CCA, introduced by Bach & Jordan [2] and generalised by Klami & Kaski [10][1]. In this section, we present an extended version of PCCA [10] (multiset PCCA[2]) which is able to handle any arbitrary number of sets. We consider a collection of datasets $\mathcal{D} = \{\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_N\}$, with each $\mathbf{X}_i \in \mathbb{R}^{D_i \times T}$ where D_i is the dimensionality and T the number of instances. By adopting the generative model for PCCA, the observation sample n of set $\mathbf{X}_i \in \mathcal{D}$ is assumed to be generated as

$$\mathbf{x}_{i,n} = f(\mathbf{z}_n|\mathbf{W}_i) + g(\mathbf{z}_{i,n}|\mathbf{B}_i) + \epsilon_i, \tag{3.1}$$

where $\mathbf{Z}_i = [\mathbf{z}_{i,1}, \ldots, \mathbf{z}_{i,T}] \in \mathbb{R}^{d_i \times T}$ and $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_T] \in \mathbb{R}^{d \times T}$ are the *independent* latent variables that capture the set-specific individual characteristics and the shared signal amongst all observation sets, respectively. $f(.)$ and $g(.)$ are functions that transform each of the latent signals $\mathbf{Z}$ and $\mathbf{Z}_i$ into the observation space. They are parametrised by $\mathbf{W}_i$ and $\mathbf{B}_i$, while the noise for each set is represented by ϵ_i, with $\epsilon_i \perp \epsilon_j, i \neq j$. Similarly to [10], $\mathbf{z}_n$, $\mathbf{z}_{i,n}$ and ϵ_i are considered to be independent (both over the set and the sequence) and normally distributed:

$$\mathbf{z}_n, \mathbf{z}_{i,n} \sim \mathcal{N}(0, \mathbf{I}), \epsilon_i \sim \mathcal{N}(0, \sigma_n^2 \mathbf{I}). \tag{3.2}$$

By considering f and g to be linear functions we have $f(\mathbf{z}_n|\mathbf{W}_i) = \mathbf{W}_i\mathbf{z}_n$ and $g(\mathbf{z}_{i,n}|\mathbf{B}_i) = \mathbf{B}_i\mathbf{z}_{i,n}$, transforming the model presented in Eq. 3.1, to

$$\mathbf{x}_{i,n} = \mathbf{W}_i\mathbf{z}_n + \mathbf{B}_i\mathbf{z}_{i,n} + \epsilon_i. \tag{3.3}$$

[1] [10] is also related to Tucker's inter-battery factor analysis [20, 4]

[2] In what follows we refer to multiset PCCA as PCCA.

Learning the multiset PCCA can be accomplished by generalising the EM algorithm presented in [10], applied to two or more sets. Firstly, $P(\mathcal{D}|\mathbf{Z}, \mathbf{Z}_1, \ldots, \mathbf{Z}_N)$ is marginalised over set-specific factors $\mathbf{Z}_1, \ldots, \mathbf{Z}_N$ and optimised on each $\mathbf{W}_i$. This leads to the generative model $P(\mathbf{x}_{i,n}|\mathbf{z}_n) \sim \mathcal{N}(\mathbf{W}_i\mathbf{z}_n, \mathbf{\Psi}_i)$, where $\mathbf{\Psi}_i = \mathbf{B}_i\mathbf{B}_i^T + \sigma_i^2\mathbf{I}$. Subsequently, $P(\mathcal{D}|\mathbf{Z}, \mathbf{Z}_1, \ldots, \mathbf{Z}_N)$ is marginalised over the common factor $\mathbf{Z}$ and then optimised on each $\mathbf{B}_i$ and σ_i. When generalising the algorithm for more than two sets, we also have to consider how to (i) obtain the expectation of the latent space and (ii) provide stable variance updates for all sets.

Two quantities are of interest regarding the latent space estimation. The first is the common latent space given one set, $\mathbf{Z}|\mathbf{X}_i$. In the classical CCA this is analogous to finding the canonical variables [10]. We estimate the posterior of the shared latent variable $\mathbf{Z}$ as follows:

$$\begin{aligned} P(\mathbf{z}_n|\mathbf{x}_{i,n}) &\sim \mathcal{N}(\boldsymbol{\gamma}_i\mathbf{x}_{i,n}, \mathbf{I} - \gamma_i\mathbf{W}_i), \\ \boldsymbol{\gamma}_i &= \mathbf{W}_i^T(\mathbf{W}_i\mathbf{W}_i^T + \mathbf{\Psi}_i)^{-1}. \end{aligned} \tag{3.4}$$

The latent space given the n-th sample from *all* sets in $\mathcal{D}$, which provides a better estimate of the shared signal manifested in all observation sets is estimated as

$$\begin{aligned} P(\mathbf{z}_n|\mathbf{x}_{1:N,n}) &\sim \mathcal{N}(\boldsymbol{\gamma}\mathbf{x}_{1:N,n}, \mathbf{I} - \boldsymbol{\gamma}\mathbf{W}), \\ \boldsymbol{\gamma} &= \mathbf{W}^T(\mathbf{W}\mathbf{W}^T + \mathbf{\Psi})^{-1}, \end{aligned} \tag{3.5}$$

while the matrices $\mathbf{W}$, $\mathbf{\Psi}$ and $\mathbf{X}_n$ are defined as $\mathbf{W}^T = [\mathbf{W}_1^T, \mathbf{W}_2^T, \ldots, \mathbf{W}_n^T]$, $\mathbf{\Psi}$ as the block diagonal matrix of $\mathbf{\Psi}_{i=1:N}$ [3] and $\mathbf{x}_{1:N,n}^T = [\mathbf{x}_{1,n}^T, \mathbf{x}_{2,n}^T, \ldots, \mathbf{x}_{1:N,n}^T]$. Finally, the variance is recovered on the full model, $x_{i,n} \sim \mathcal{N}(\mathbf{W}_i\mathbf{z}_n + \mathbf{B}_i\mathbf{z}_{i,n}, \sigma_i^2\mathbf{I})$, as

$$\begin{aligned} \sigma_i^2 =& tr(\mathbf{S} - \mathbf{X}\mathbb{E}[\mathbf{Z}^T|\mathbf{X}]\mathbf{C}^T \\ & -\mathbf{C}\mathbb{E}[\mathbf{Z}|\mathbf{X}]\mathbf{X}^T - \mathbf{C}\mathbb{E}[\mathbf{Z}\mathbf{Z}^T|\mathbf{X}]\mathbf{C}^T)_i\frac{T}{D_i}, \end{aligned} \tag{3.6}$$

where $\mathbf{S}$ is the sample covariance matrix, $\mathbf{B}$ is the block diagonal matrix of $\mathbf{B}_{i=1:N}$, $\mathbf{C} = [\mathbf{W}, \mathbf{B}]$, while the subscript i in Eq. 3.6 refers to the i-th block of the full covariance matrix. Finally, we note that the computational complexity of PCCA for each iteration is similar to deterministic CCA (cubic in the dimensionalities of the datasets and linear in the number of samples). PCCA though also recovers the private space.

3.2 Dynamic PCCA (DPCCA)

The PCCA model described in Sec. 3.1 exhibits several advantages when compared to the classical formulation of CCA, mainly by providing a probabilistic estimation of a latent space shared by an arbitrary collection of datasets along with explicit noise

[3] For brevity of notation, we use $1 : N$ to indicate elements $[1, \ldots, N]$, e.g., $\mathbf{X}_{1:N} \equiv [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_N]$

and private space estimation. Nevertheless, static models are unable to learn temporal dependencies which are very likely to exist when dealing with real-life problems. In fact, dynamics are deemed essential for successfully performing tasks such as emotion recognition, AU detection etc. [23].

Motivated by the former observation, we use a dynamic generalisation of the static PCCA model introduced in the previous section, where we now treat each $\mathbf{X}_i$ as a temporal sequence. For simplicity of presentation, we introduce a linear model[4] where Markovian dependencies are learnt in the latent spaces $\mathbf{Z}$ and $\mathbf{Z}_i$. In other words, the variable $\mathbf{Z}$ models the temporal, shared signal amongst all observation sequences, while $\mathbf{Z}_i$ captures the temporal, individual characteristics of each sequence. It is easy to observe that such a model fits perfectly with the problem of fusing multiple annotations, as it does not only capture the temporal shared signal of all annotations, but also models the unwanted, annotator-specific factors over time. Essentially, instead of directly applying the doubly independent priors to $\mathbf{Z}$ as in Eq. 3.2, we now use the following:

$$p(\mathbf{z}_t|\mathbf{z}_{t-1}) \sim \mathcal{N}(\mathbf{A}_z\mathbf{z}_{t-1}, \mathbf{V}_Z), \tag{3.7}$$

$$p(\mathbf{z}_{i,t}|\mathbf{z}_{i,t-1}) \sim \mathcal{N}(\mathbf{A}_{z_i}\mathbf{z}_{i,t-1}, \mathbf{V}_{Z_i}), n = 1, \ldots, N, \tag{3.8}$$

where the transition matrices $\mathbf{A}_z$ and $\mathbf{A}_{z_i}$ model the latent space dynamics for the shared and sequence-specific space respectively. Thus, idiosyncratic characteristics of dynamic nature appearing in a single sequence can be accurately estimated and prevented from contaminating the estimation of the shared signal.

The resulting model bears similarities with traditional Linear Dynamic System (LDS) models (e.g. [17]) and the so-called Factorial Dynamic Models, c.f. [5]. Along with Eq. 3.7,3.8 and noting Eq. 3.3, the dynamic, generative model for DPCCA[5] can be described as

$$\mathbf{x}_{i,t} = \mathbf{W}_{i,t}\mathbf{z}_t + \mathbf{B}_i\mathbf{z}_{i,t} + \epsilon_i, \epsilon_i \sim \mathcal{N}(0, \sigma_i^2\mathbf{I}), \tag{3.9}$$

where the subscripts i and t refer to the i-th observation sequence timestep t respectively.

3.2.1 Inference

To perform inference, we reduce the DPCCA model to a LDS[6]. This can be accomplished by defining a joint space $\hat{\mathbf{Z}}^T = [\mathbf{Z}^T, \mathbf{Z}_1^T, \ldots, \mathbf{Z}_N^T]$, $\hat{\mathbf{Z}} \in \mathbb{R}^{\hat{d}\times T}$ where $\hat{d} = d + \sum_i^N d_i$ with parameters $\boldsymbol{\theta} = \{\mathbf{A}, \mathbf{W}, \mathbf{B}, \mathbf{V}_{\hat{z}}, \hat{\boldsymbol{\Sigma}}\}$. Dynamics in this joint space are described as $\mathbf{X}_t = [\mathbf{W}, \mathbf{B}]\hat{\mathbf{Z}}_t + \boldsymbol{\epsilon}, \hat{\mathbf{Z}}_t = \mathbf{A}\hat{\mathbf{Z}}_{t-1} + \mathbf{u}$, where the noise processes $\boldsymbol{\epsilon}$ and $\mathbf{u}$

[4] A non-linear DPCCA model can be derived similarly to [9, 6].

[5] The model of Raykar et al. [16] can be considered as a special case of (D)PCCA by setting $\mathbf{W} = \mathbf{I}$, $\mathbf{B} = \mathbf{0}$ (and disregarding dynamics).

[6] For more details on LDS, please see [17] and [3], Chapter 13.

are defined as

$$\boldsymbol{\epsilon} \sim \mathcal{N}\left(0, \underbrace{\begin{bmatrix} \sigma_1^2\mathbf{I} & & \\ & \ddots & \\ & & \sigma_N^2\mathbf{I} \end{bmatrix}}_{\hat{\boldsymbol{\Sigma}}}\right), \tag{3.10}$$

$$\mathbf{u} \sim \mathcal{N}\left(0, \underbrace{\begin{bmatrix} \mathbf{V}_z & & & \\ & \mathbf{V}_{z_1} & & \\ & & \ddots & \\ & & & \mathbf{V}_{z_N} \end{bmatrix}}_{\mathbf{v}_{\hat{z}}}\right), \tag{3.11}$$

where $\mathbf{V}_z \in \mathbb{R}^{d \times T}$ and $\mathbf{V}_{z_i} \in \mathbb{R}^{d_i \times T}$. The other matrices used above are defined as $\mathbf{X}^T = [\mathbf{X}_1^T, \ldots, \mathbf{X}_N^T]$, $\mathbf{W}^T = [\mathbf{W}_1^T, \ldots, \mathbf{W}_N^T]$, $\mathbf{B}$ as the block diagonal matrix of $[\mathbf{B}_1, \ldots, \mathbf{B}_N]$ and $\mathbf{A}$ as the block diagonal matrix of $[\mathbf{A}_z, \mathbf{A}_{z_1}, \ldots, \mathbf{A}_{z_N}]$. Similarly to LDS, the joint log-likelihood function of DPCCA is defined as

$$\begin{aligned} lnP(\mathbf{X}, \mathbf{Z}|\theta) =& lnP(\hat{\mathbf{z}}_1|\mu, V) + \sum_{t=2}^{T} lnP(\hat{\mathbf{z}}_t|\hat{\mathbf{z}}_{t-1}, \mathbf{A}, \mathbf{V}_{\hat{z}}) \\ &+ \sum_{t=1}^{T} lnP(\mathbf{x}_t|\hat{\mathbf{z}}_t, \mathbf{W}, \mathbf{B}, \hat{\boldsymbol{\Sigma}}). \end{aligned} \tag{3.12}$$

In order estimate the latent spaces, we apply the Rauch-Tung-Striebel (RTS) smoother on $\hat{\mathbf{Z}}$ (the algorithm can be found in [17], A.3). In this way, we obtain $\mathbb{E}[\hat{\mathbf{z}}_t|\mathbf{X}^T]$, $V[\hat{\mathbf{z}}_t|\mathbf{X}^T]$ and $V[\hat{\mathbf{z}}_t\hat{\mathbf{z}}_{t-1}|\mathbf{X}^T]$[7].

3.2.2 Parameter Estimation

The parameter estimation of the M-step has to be derived specifically for this factorised model. We consider the expectation of the joint model log-likelihood (Eq. 3.12) wrt. posterior and obtain the partial derivatives of each parameter for finding the sta-

[7] We note that the complexity of RTS is cubic in the dimension of the state space. Thus, when estimating high dimensional latent spaces, computational or numerical issues may arise (due to the inversion of large matrices). If any of the above is a concern, the complexity of RTS can be reduced to quadratic [21], while inference can be performed more efficiently similarly to [5].

tionary points. Note the $\mathbf{W}$ and $\mathbf{B}$ matrices appear in the likelihood as:

$$\mathbb{E}_{\hat{z}}[lnP(\mathbf{X}, \hat{\mathbf{Z}})] = -\frac{T}{2} ln|\hat{\mathbf{\Sigma}}| - \mathbb{E}_{\hat{z}}\Bigg[\sum_{t=1}^{T} (\mathbf{x}_t - [\mathbf{W}, \mathbf{B}]\hat{\mathbf{z}}_t)^T \hat{\mathbf{\Sigma}}^{-1} (\mathbf{x}_t - [\mathbf{W}, \mathbf{B}]\hat{\mathbf{z}}_t)\Bigg] + \dots. \tag{3.13}$$

Since they are composed of individual $\mathbf{W}_i$ and $\mathbf{B}_i$ matrices (which are parameters for each sequence i), we calculate the partial derivatives $\partial\mathbf{W}_i$ and $\partial\mathbf{B}_i$ in Eq. 3.13. Subsequently, by setting to zero and re-arranging, we obtain the update equations for each $\mathbf{W}_i^*$ and $\mathbf{B}_i^*$:

$$\mathbf{W}_i^* = \left(\sum_{t=1}^{T} \mathbf{x}_{i,t}\mathbb{E}[\mathbf{z}_{i,t}] - \mathbf{B}_i^*\mathbb{E}[\mathbf{z}_{i,t}\mathbf{z}_t^T]\right)\left(\sum_{t=1}^{T} \mathbb{E}[\mathbf{z}_t\mathbf{z}_t^T]\right)^{-1} \tag{3.14}$$

$$\mathbf{B}_i^* = \left(\sum_{t=1}^{T} \mathbf{x}_{i,t}\mathbb{E}[\mathbf{z}_t^T] - \mathbf{W}_i^*\mathbb{E}[\mathbf{z}_t\mathbf{z}_{i,t}^T]\right)\left(\sum_{t=1}^{T} \mathbb{E}[\mathbf{z}_{i,t}\mathbf{z}_{i,t}^T]\right)^{-1} \tag{3.15}$$

Note that the weights are *coupled* and thus the optimal solution should be found iteratively. As can be seen, in contrast to PCCA, in DPCCA the individual factors of each sequence are explicitly estimated instead of being marginalised out. Similarly, the transition weight updates for the individual factors $\mathbf{Z}_i$ are as follows:

$$\mathbf{A}_{z,i}^* = \left(\sum_{t=2}^{T} E[\mathbf{z}_{i,t}\mathbf{z}_{i,t-1}^T]\right)\left(\sum_{t=2}^{T} E[\mathbf{z}_{i,t-1}\mathbf{z}_{i,t-1}^T]\right)^{-1} \tag{3.16}$$

where by removing the subscript i we obtain the updates for $\mathbf{A}_z$, corresponding to the shared latent space $\mathbf{Z}$. Finally, the noise updates $\mathbf{V}_{\hat{Z}}$ and $\hat{\mathbf{\Sigma}}$ are estimated similarly to LDS [17].

3.3 DPCCA with Time Warpings

Both PCCA and DPCCA exhibit several advantages in comparison to the classical formulation of CCA. Mainly, as we have shown, (D)PCCA can inherently handle more than two sequences, building upon the multiset nature of PCCA. This is in contrast to the classical formulation of CCA, which due to the pairwise nature of the correlation operator is limited to two sequences[8]. This is crucial for the problems at hand since both methods yield an accurate estimation of the underlying signals of *all* observation sequences, free of individual factors and noise. However, both PCCA and DPCCA carry the assumption that the temporal correspondences between samples of different

[8]The recently proposed multiset-CCA [8] can handle multiple sequences but requires maximising over sums of pairwise operations.

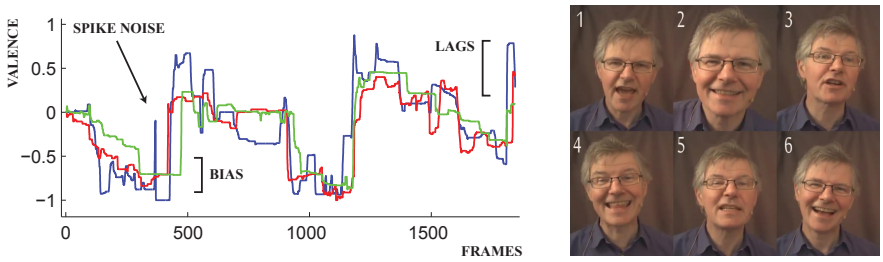

Figure 3.1: Valence annotations along with video stills.

sequences are *known*, i.e. that the annotation of expert i at time t directly corresponds to the annotation of expert j at the same time. Nevertheless, this assumption is often violated since different experts exhibit different time lags in annotating the same process (e.g., Fig. 3.1, [12]). Motivated by the latter, we extend the DPCCA model to account for this *misalignment* of data samples by introducing a latent warping process into DPCCA, in a manner similar to [25]. In what follows, we firstly describe some basic background on time-warping and subsequently proceed to define our model.

3.3.1 Time Warping

Dynamic Time Warping (DTW) [15] is an algorithm for optimally aligning two sequences of possibly different lengths. Given sequences $\mathbf{X} \in \mathbb{R}^{D \times T_x}$ and $\mathbf{Y} \in \mathbb{R}^{D \times T_y}$, DTW aligns the samples of each sequence by minimising the sum-of-squares cost, i.e. $||\mathbf{X}\boldsymbol{\Delta}_x - \mathbf{Y}\boldsymbol{\Delta}_y||_F^2$, where $\boldsymbol{\Delta}_x \in \mathbb{R}^{T_x \times T_\Delta}$ and $\boldsymbol{\Delta}_y \in \mathbb{R}^{T_y \times T_\Delta}$ are binary selection matrices, with T_Δ the aligned, common length. In this way, the warping matrices $\boldsymbol{\Delta}$ effectively re-map the samples of each sequence. Although the number of possible alignments is exponential in $T_x T_y$, employing dynamic programming can recover the optimal path in $\mathcal{O}(T_x T_y)$. Furthermore, the solution must satisfy the boundary, continuity and monotonicity constraints, effectively restricting the space of $\boldsymbol{\Delta}_x$, $\boldsymbol{\Delta}_y$ [15].

An important limitation of DTW is the inability to align signals of different dimensionality. Motivated by the former, CTW [25] combines CCA and DTW, thus alowing the alignment of signals of different dimensionality by projecting into a common space via CCA. The optimisation function now becomes $||\mathbf{V}_x^T \mathbf{X}\boldsymbol{\Delta}_x - \mathbf{V}_y^T \mathbf{Y}\boldsymbol{\Delta}_y||_F^2$, where $\mathbf{X} \in \mathbb{R}^{D_x \times T_x}$, $\mathbf{Y} \in \mathbb{R}^{D_y \times T_x}$, and $\mathbf{V}_x, \mathbf{V}_y$ are the projection operators (matrices).

3.3.2 DPCTW Model

We define DPCTW based on the graphical model presented in Fig. 3.2. Given a set $\mathcal{D}$ of N sequences of varying duration, with each sequence $\mathbf{X}_i = [\mathbf{x}_{i,1}, \ldots, \mathbf{x}_{i,T_i}] \in \mathbb{R}^{D_i \times T_i}$, we postulate the latent common Markov process $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_t\}$. Firstly, $\mathbf{Z}$ is warped using the warping operator $\boldsymbol{\Delta}_i$, resulting in the warped latent sequence $\boldsymbol{\zeta}_i$. Subsequently, each $\boldsymbol{\zeta}_i$ generates each observation sequence $\mathbf{X}_i$, also considering the

annotator/sequence bias $\mathbf{Z}_i$ and the observation noise σ_i^2. We note that we do not impose parametric models for warping processes. Inference in this general model can be prohibitively expensive, in particular because of the need to handle the unknown alignments. We instead decided to handle the inference in two steps: (i) fix the alignments $\mathbf{\Delta}_i$ and find the latent $\mathbf{Z}$ and $\mathbf{Z}_i$'s, and (ii) given the estimated $\mathbf{Z}, \mathbf{Z}_i$ find the optimal warpings $\mathbf{\Delta}_i$. For this, we decided to optimise the following objective function:

$$\mathcal{L}_{\text{(D)PCTW}} = \sum_{i}^{N} \sum_{j,j\neq i}^{N} \frac{||\mathbb{E}[\mathbf{Z}|\mathbf{X}_i]\mathbf{\Delta}_i - \mathbb{E}[\mathbf{Z}|\mathbf{X}_j]\mathbf{\Delta}_j||_F^2}{N(N-1)} \tag{3.17}$$

where when using PCCA, $\mathbb{E}[\mathbf{Z}|\mathbf{X}_i] = \mathbf{W}_i^T(\mathbf{W}_i\mathbf{W}_i^T + \mathbf{\Psi}_i)^{-1}\mathbf{X}_i$ (Eq. 3.4). For DPCCA, $\mathbb{E}[\mathbf{Z}|\mathbf{X}_i]$ is inferred via RTS smoothing (Sec. 3.2). A summary of the full algorithm is presented in Algorithm 1.

At this point, it is important to clarify that our model is flexible enough to be straightforwardly used with varying warping techniques. For example, the Gauss-Newton warping proposed in [24] can be used as the underlying warping process for DPCCA, by replacing the projected data $\mathbf{V}_i^T\mathbf{X}_i$ with $\mathbb{E}[\mathbf{Z}|\mathbf{X}_i]$ in the optimisation function. Algorithmically, this only changes the warping process (line 3, Algorithm 1). Finally, we note that since our model iterates between estimating the latent spaces with (D)PCCA and warping, the computational complexity of time warping is additive to the cost of each iteration. In case of the DTW alignment for two sequences, this incurs an extra cost of $O(T_xT_y)$. In case of more than two sequences, we utilise a DTW-based algorithm, which is a variant of the so-called Guide Tree Progressive Alignment, since the complexity of dynamic programming increases exponentially with the number of sequences. Similar algorithms are used in state-of-the-art sequence alignment software in biology, e.g., Clustar [11]. The complexity of the employed algorithm is $O(N^2T_{max}^2)$ where T_{max} is the maximum (aligned) sequence length and N the number of sequences. More efficient implementations can also be used by employing various constraints [15].

3.4 Features for Annotator Fusion

In the previous sections, we considered the observed data to consist only of the given annotations, $\mathcal{D} = \{\mathbf{X}_1, \ldots, \mathbf{X}_N\}$. Nevertheless, in many problems one can extract additional observed information, which we can consider as a form of *complementary input* (e.g., visual or audio features). In fact, in problems where annotations are subjective and no objective ground truth is available for any portion of the data, such input can be considered as the only objective reference to the annotation/sequence at hand. Thus, incorporating it into the model can significantly aid the determination of the ground truth.

Motivated by the latter argument, we used two models which augment DPCCA/ DPCTW with inputs. Since the family of component analysis techniques we study

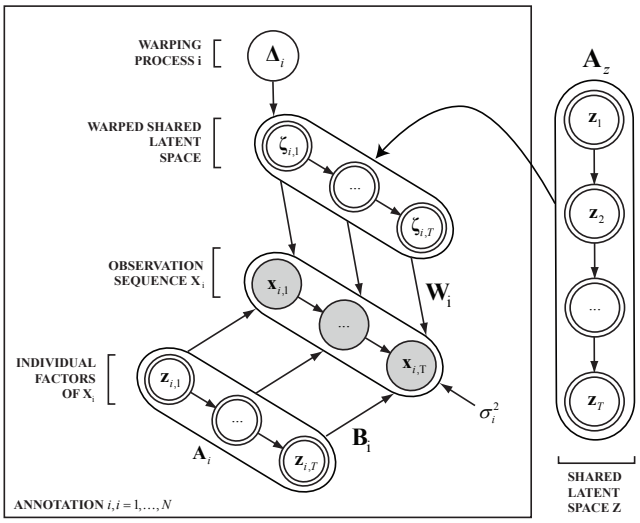

Figure 3.2: Graphical model of DPCTW. Shaded nodes represent the observations. By ignoring the temporal dependencies, we obtain the PCTW model.

Algorithm 1: Dynamic Probabilistic CCA with Time Warpings (DPCTW)

Data: $\mathcal{D} = \mathbf{X}_1, \ldots, \mathbf{X}_N,\ \mathbf{X}^T = [\mathbf{X}_1^T, \ldots, \mathbf{X}_N^T]$
Result: $P(\mathbf{Z}|\mathbf{X}_1, \ldots \mathbf{X}_N), P(\mathbf{Z}|\mathbf{X}_i), \boldsymbol{\Delta}_i, \sigma_i^2, i = 1:N$

repeat
  Obtain alignment matrices $(\boldsymbol{\Delta}_1, \ldots, \boldsymbol{\Delta}_N)$ by optimising Eq. 3.17 on $\mathbb{E}[\mathbf{Z}|\mathbf{X}_1^T], \ldots, \mathbb{E}[\mathbf{Z}|\mathbf{X}_N^T]$*
  $\mathbf{X}_\Delta^T = [(\mathbf{X}_1\boldsymbol{\Delta}_1)^T, \ldots, (\mathbf{X}_N\boldsymbol{\Delta}_N)^T]$
  repeat
    Estimate $\mathbb{E}[\hat{\mathbf{z}}_t|\mathbf{X}_\Delta^T]$, $V[\hat{\mathbf{z}}_t|\mathbf{X}_\Delta^T]$ and $V[\hat{\mathbf{z}}_t\hat{\mathbf{z}}_{t-1}|\mathbf{X}_\Delta^T]$ via RTS
    for $i = 1, \ldots, N$ **do**
      repeat
        Update $\mathbf{W}_i^*$ according to Eq. 3.14
        Update $\mathbf{B}_i^*$ according to Eq. 3.15
      until $\mathbf{W}_i$, $\mathbf{B}_i$ *converge*
      Update $\mathbf{A}_i^*$ according to Eq. 3.16
    Update $\mathbf{A}^*, \mathbf{V}_{\hat{Z}}^*, \hat{\boldsymbol{\Sigma}}^*$ according to Sec. 3.2.2
  until *DPCCA converges*
  for $i = 1, \ldots, N$ **do**
    $\boldsymbol{\theta}_i = \left\{ \begin{bmatrix} \mathbf{A}_z & 0 \\ 0 & \mathbf{A}_i \end{bmatrix}, \mathbf{W}_i, \mathbf{B}_i, \begin{bmatrix} \mathbf{V_Z} & 0 \\ 0 & \mathbf{V}_i \end{bmatrix}, \sigma_i^2\mathbf{I} \right\}$
    Estimate $\mathbb{E}[\hat{\mathbf{z}}_t|\mathbf{X}_i^T]$, $V[\hat{\mathbf{z}}_t|\mathbf{X}_i^T]$ and $V[\hat{\mathbf{z}}_t\hat{\mathbf{z}}_{t-1}|\mathbf{X}_i^T]$ via RTS on θ_i.
until $\mathcal{L}_{DPCTW}$ *converges*

* Since $\mathbb{E}[\hat{\mathbf{z}}_t|\mathbf{X}_i^T]$ is unkown in the first iteration, use $\mathbf{X}_i$ instead.

are typically unsupervised, incorporating inputs leads to a form of supervised learning. Such models can find a wide variety of applications since they are able to exploit label information in addition to observations. A suitable example lies in dimensional affect analysis, where it has been shown that specific emotion dimensions correlate better with specific cues, (e.g., valence with facial features, arousal with audio features [13, 7]). Thus, one can know a-priori which features to use for specific annotations.

Throughout this discussion, we assume that a set of complementary input or features $\mathbf{Y} = \{\mathbf{Y}_1, \ldots, \mathbf{Y}_\nu\}$ is available, where $\mathbf{Y}_j \in \mathbb{R}^{D_{y_j} \times T_{y_j}}$. While discussing extensions of DPCCA, we assume that all sequences have equal length. When incorporating time warping, sequences can have different lengths.

3.4.1 Supervised-Generative DPCCA (SG-DPCCA)

We firstly consider the model where we simply augment the observation model with a set of features $\mathbf{Y}_j$. In this case, the generative model for DPCCA (Eq. 3.9) is:

$$\mathbf{x}_{i,t} = \mathbf{W}_{i,t}\mathbf{z}_t + \mathbf{B}_i\mathbf{z}_{i,t} + \epsilon_i, \tag{3.18}$$

$$\mathbf{y}_{j,t} = h_{j,s}(\mathbf{z}_t|\mathbf{W}_{j,t}) + h_{j,p}(\mathbf{z}_{j,t}|\mathbf{B}_j) + \epsilon_j, \tag{3.19}$$

where $i = \{1, \ldots, N\}$ and $j = \{N+1, \ldots, N+\nu+1\}$. The arbitrary functions h map the shared space to the feature space in a generative manner, while $\epsilon_j \sim \mathcal{N}(0, \sigma_j^2\mathbf{I})$. The latent priors are still defined as in Eq. 3.7,3.8. By assuming that h is linear, we can group the parameters $\mathbf{W} = [\mathbf{W}_1, \ldots, \mathbf{W}_N, \ldots, \mathbf{W}_{N+\nu}]$, $\mathbf{B}$ as the block diagonal of $([\mathbf{B}_1, \ldots, \mathbf{B}_N, \ldots, \mathbf{B}_{N+\nu}])$ and $\hat{\mathbf{\Sigma}}$ as the block diagonal of $([\sigma^2\mathbf{I}_1, \ldots, \sigma^2\mathbf{I}_N, \ldots, \sigma^2\mathbf{I}_{N+\nu}])$. Inference is subsequently applied as described in Sec. 3.2.

This model, which we dub SG-DPCCA, in effect captures a common shared space of both annotations $\mathbf{X}$ and available features $\mathbf{Y}$ for each sequence. In our generative scenario, the shared space generates both features and annotations. By further setting $h_{j,p}$ to zero, one can force the representation of the entire feature space $\mathbf{Y}_j$ onto the shared space, thus imposing stronger constraints on the shared space given each annotation $\mathbf{Z}|\mathbf{X}_i$. As we will show, this model can help identify unwanted annotations by simply analysing the posteriors of the shared latent space. We note that the additional form of supervision imposed by the input on the model is reminiscent of SPCA for PCA [22]. The discriminative ability added by the inputs (or labels) also relates DPCCA to LDA [2]. The graphical model of SG-DPCCA is illustrated in Fig. 3.3(b).

SG-DPCCA can be easily extended to handle time-warping as described in Sec. 3.3 for DPCCA (SG-DPCTW). The main difference is that now one would have to introduce one more warping function for each set of features, resulting in a set of $N+\nu$ functions. Denoting the complete data/input set as $\mathcal{D}^o = \{\mathbf{X}_1, \ldots, \mathbf{X}_N, \mathbf{Y}_1, \ldots, \mathbf{Y}_\nu\}$, the objective

function for obtaining the time warping functions Δ_i for SG-DPCTW can be defined as:

$$\mathcal{L}_{SDPCTW^o} = \sum_{i}^{N+\nu} \sum_{j,j\neq i}^{N+\nu} \frac{||\mathbb{E}[\mathbf{Z}|\mathcal{D}_i^o]\mathbf{\Delta}_i - \mathbb{E}[\mathbf{Z}|\mathcal{D}_j^o]\mathbf{\Delta}_j||_F^2}{(N+\nu)(N+\nu-1)}. \tag{3.20}$$

3.4.2 Supervised-Discrimative DPCCA (SD-DPCCA)

The second model augments the DPCCA model by regressing on the given features. In this case, the posterior of the shared space (Eq. 3.7) is formulated as

$$p(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{Y}_{1:\nu}, \mathbf{A}, \mathbf{V}_{\hat{z}}) \sim \mathcal{N}(\mathbf{A}_z\mathbf{z}_{t-1} + \sum_{j=1}^{\nu} h_j(\mathbf{Y}_j|\mathbf{F}_j), \mathbf{V}_z), \tag{3.21}$$

where each function h_j performs regression on the features $\mathbf{Y}_j$, while $\mathbf{F}_j \in \mathbb{R}^{d\times D_{y_j}}$ are the loadings for the features (where the latent dimensionality is d). This is similar to how input is modelled in a standard LDS [6]. To find the parameters, we maximise the complete-data likelihood (Eq. 3.12), where we replace the second term referring to the latent probability with Eq. 3.21,

$$\sum_{t=2}^{T} lnP(\hat{\mathbf{z}}_t|\hat{\mathbf{z}}_{t-1}, \mathbf{Y}_{1:\nu}, \mathbf{A}, \mathbf{V}_{\hat{z}}). \tag{3.22}$$

In this variation, the shared space at step t is generated from the previous latent state $\mathbf{z}_{t-1}$ as well as the features at step $t-1, \sum_{j=1}^{\nu} \mathbf{y}_{j,t-1}$ (Fig. 3.3(c)). We dub this model SD-DPCCA. Without loss of generality we assume h is linear, i.e. $h_{j,s} = \mathbf{W}_{j,t}\mathbf{z}_t$, while we model the feature signal only in the shared space, i.e. $h_{j,p} = 0$. Finding the saddle points of the derivatives with respect to the parameters yields the following updates for the matrices $\mathbf{A}_z$ and $\mathbf{F}_j, \forall j = 1, \ldots, \nu$:

$$\mathbf{A}_z^* = \left(\sum_{t=2}^{T} E[\mathbf{z}_t\mathbf{z}_{t-1}^T] - \sum_{j=1}^{\nu} \mathbf{F}_j^*\mathbf{y}_{j,t}\right)\left(\sum_{t=2}^{T} E[\mathbf{z}_{t-1}\mathbf{z}_{t-1}^T]\right)^{-1}, \tag{3.23}$$

$$\mathbf{F}_j^* = \left(\mathbb{E}[\mathbf{z}_t] - \mathbf{A}_z^*\mathbb{E}[\mathbf{z}_{t-1}] - \sum_{i=1,i\neq j}^{\nu} \mathbf{F}_i^*\mathbf{Y}_i\right)\mathbf{Y}_j^{-1}. \tag{3.24}$$

Note that as with the loadings on the shared/individual spaces ($\mathbf{W}$ and $\mathbf{B}$), the optimisation of $\mathbf{A}_z$ and $\mathbf{F}_j$ matrices should again be determined recursively. Finally, the estimation of $\mathbf{V_Z}$ also changes accordingly:

$$\begin{aligned}\mathbf{V}_\mathbf{z}^* = \; & \tfrac{1}{T-1}\textstyle\sum_{t=2}^{T}(\mathbb{E}[\mathbf{z}_t\mathbf{z}_t^T] - \mathbb{E}[\mathbf{z}_t\mathbf{z}_{t-1}^T]\mathbf{A}_z^{*T} \\ & -\mathbf{A}_z^*\mathbb{E}[\mathbf{z}_{t-1}\mathbf{z}_t^T] + \mathbf{A}_z^*\mathbb{E}[\mathbf{z}_{t-1}\mathbf{z}_{t-1}^T]\mathbf{A}_z^{*T} \\ & +\textstyle\sum_{j=1}^{\nu}(\mathbf{A}_z^*\mathbb{E}[\mathbf{z}_{t-1}]\mathbf{Y}_j^{*T}\mathbf{F}_j^{*T} + \mathbf{F}_j^*\mathbf{Y}_j\mathbb{E}[\mathbf{z}_{t-1}^T]\mathbf{A}_z^{*T} \\ & +\mathbf{F}_j^*\mathbf{Y}_j\textstyle\sum_{i=1,i\neq j}^{\nu}\mathbf{Y}_i^T\mathbf{F}_i^{*T} - \mathbb{E}[\mathbf{z}_t]\mathbf{Y}_j^T\mathbf{F}_j^{*T} \\ & -\mathbf{F}_j^*\mathbf{Y}_j\mathbb{E}[\mathbf{z}_t^T])).\end{aligned} \tag{3.25}$$

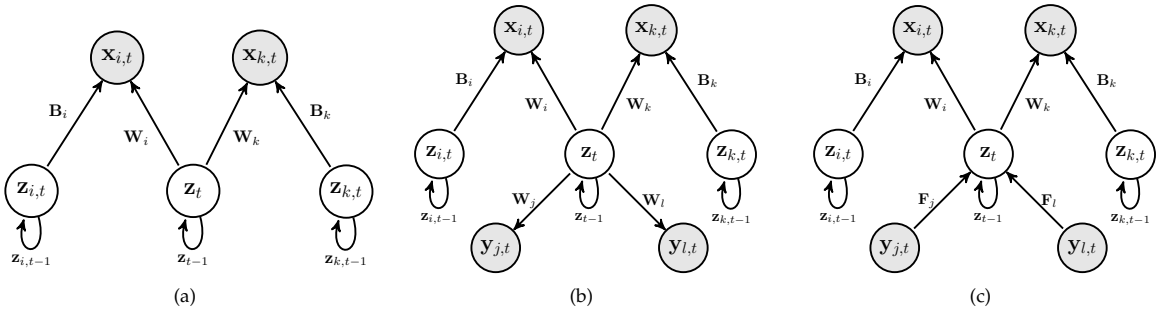

Figure 3.3: Comparing the model structure of DPCCA (a) to SG-DPCCA (b) and SD-DPCCA (c). Notice that the shared space $\mathbf{z}$ generates both observations and features in SG-DPCCA, while in SD-DPCCA, the shared space at time t is generated by regressing from the features $\mathbf{y}$ and the previous shared space state $\mathbf{z}_{t-1}$.

SD-DPCCA can be straight-forwardly extended with time-warping as with DPCCA in Sec. 3.3, resulting in SD-DPCTW. Another alignment step is required before performing the recursive updates mentioned above in order to find the correct training/testing pairs for $\mathbf{z}_t$ and $\mathbf{Y}$. Assuming the warping matrices are $\mathbf{\Delta}_z$ and $\mathbf{\Delta}_y$, then in Eq. 3.23 $\mathbf{z}$ is replaced with $\mathbf{\Delta}_z\mathbf{z}$ and $\mathbf{y}$ with $\mathbf{\Delta}_y\mathbf{y}$. The influence of features $\mathbf{Y}$ on the shared latent space $\mathbf{Z}$ in SD-DPCCA and SG-DPCCA is visualised in Fig. 3.3.

3.4.3 Varying Dimensionality

Typically, we would expect the dimensionality of a set of annotations to be the same. Nevertheless in certain problems, especially when using input features as in SG-DPCCA (Sec. 3.4.1), this is not the case. Therefore, in case the observations/input features are of varying dimensionalities, one can scale the third term of the likelihood (Eq. 3.12) in order to balance the influence of each sequence during learning regardless of its dimensionality:

$$\sum_{t=1}^{T}\left(\sum_{j=1}^{\nu}\frac{1}{D_{y_j}}ln\left(P(\mathbf{y}_{t,j}|\hat{\mathbf{z}}_t,\mathbf{W}_j,\mathbf{B}_j,\sigma_j^2)\right)+\sum_{j=1}^{N}\frac{1}{D_i}ln\left(P(\mathbf{x}_{t,j}|\hat{\mathbf{z}}_t,\mathbf{W}_j,\mathbf{B}_j,\sigma_i^2)\right)\right). \tag{3.26}$$

3.5 Ranking and filtering annotations

In this section, we will refer to the issue of ranking and filtering available annotations. Since in general, we consider that there is no "ground truth" available, it is not an easy task to infer which annotators should be discarded and which kept. A straightforward option would be to keep the set of annotators which exhibit a decent level of agreement

with each other. Nevertheless, this naive criterion will not suffice in case where e.g., all the annotations exhibit moderate correlation, or where sets of annotations are clustered in groups which are intra-correlated but not inter-correlated.

The question that naturally arises is how to rank and evaluate the annotators when there is no ground truth available and their inter-correlation is not helpful. We remind that DPCCA maximises the correlation of the annotations in the shared space $\mathbf{Z}$, by removing bias, temporal discrepancies and other nuisances from each annotation. It would therefore be reasonable to expect the latent *posteriors* for each annotation ($\mathbf{Z}|\mathbf{X}_i$), to be as close as possible. Furthermore, the closer the posterior given each annotation ($\mathbf{Z}|\mathbf{X}_i$) to the posterior given all sequences ($\mathbf{Z}|\mathcal{D}$), the higher the ranking of the annotator should be, since the closer it is, the larger the portion of the shared information is contained in the annotators signal.

The aforementioned procedure can detect *spammers*, i.e. annotators who do not even pay attention at the sequence they are annotating and *adversarial* or *malicious* annotators that provide erroneous annotations due to e.g., a conflict of interests and can rank the confidence that should be assigned to the rest of the annotators. Nevertheless, it does not account for the case where multiple clusters of annotators are intra-correlated but not inter-correlated. In this case, it is most probable that the best-correlated group will prevail in the ground truth determination. Yet, this does not mean that the best-correlated group is the correct one. In this case, we use a set of inputs (e.g., tracking facial points), which can essentially represent the "gold standard". The assumption underlying this proposal is that the correct sequence features should maximally correlate with the correct annotations of the sequence. This can be straightforwardly performed with SG-DPCCA, where we attain $\mathbf{Z}|\mathbf{Y}$ (shared space given input) and compare to $\mathbf{Z}|\mathbf{X}_i$ (shared space given annotation i).

The comparison of latent posteriors is further motivated by R.J. Aumann's agreement theorem [1]: "If two people are Bayesian rationalists with common priors, and if they have common knowledge of their individual posteriors, then their posteriors must be equal". Since our model maintains the notion of "common knowledge" in the estimation of the shared space, it follows from Aumann's theorem that the individual posteriors $\mathbf{Z}|\mathbf{X}_i$ of each annotation i should be as close as possible. This is a sensible assumption, since one would expect that if all bias, temporal discrepancies and other nuisances are removed from annotations, then there is no rationale for the posteriors of the shared space to differ.

A simple algorithm for filtering/ranking annotations (utilising spectral clustering [18]) can be found in Algorithm 2. The goal of the algorithm is to find two clusters, C_x and C_o, containing (i) the set of annotations which are correlated with the ground truth, and (ii) the set of "outlier" annotations, respectively. Firstly, DPCCA/DPCTW is applied. Subsequently, a similarity/distance matrix is constructed based on the posterior

distances of each annotation $\mathbf{Z}|\mathbf{X}_i$ along with the features $\mathbf{Z}|\mathbf{Y}$. By performing spectral clustering, one can keep the cluster to which $\mathbf{Z}|\mathbf{Y}$ belongs (C_x) and disregard the rest of the annotations belonging in C_o. The ranking of the annotators is computed implicitly via the distance matrix, as it is the relative distance of each $\mathbf{Z}|\mathbf{X}_i$ to $\mathbf{Z}|\mathbf{Y}$. In other words, the feature posterior is used here as the "ground truth". Depending on the application (or in case features are not available), one can use the posterior given all annotations, $\mathbf{Z}|\mathbf{X}_1, \ldots, \mathbf{X}_N$ instead of $\mathbf{Z}|\mathbf{Y}$. Examples of distances/metrics that can be used include the alignment error (see Sec. 3.3) or the KL divergence between normal distributions (which can be made symmetric by employing e.g., the Jensen-Shannon divergence, i.e. $D_{JS}(P||Q) = \frac{1}{2}D_{KL}(P||Q) + \frac{1}{2}D_{KL}(Q||P)$).

Algorithm 2: Ranking and filtering annotators

Data: $\mathbf{X}_1, \ldots, \mathbf{X}_N, \mathbf{Y}$
Result: Rank of each $\mathbf{X}_i$, C_c
begin
  Apply SG-DPCTW/SG-DPCCA($\mathbf{X_1}, \ldots, \mathbf{X_N}, \mathbf{Y}$)
  Obtain $P(\mathbf{Z}|\mathbf{Y})$, $P(\mathbf{Z}|\mathbf{X}_i), i = 1, \ldots, N$
  Compute Distance Matrix $\mathbf{S}$ of $[P(\mathbf{Z}|\mathbf{X}_1), \ldots, P(\mathbf{Z}|\mathbf{X}_N), P(\mathbf{Z}|\mathbf{Y})]$
  Normalise $\mathbf{S}$, $\mathbf{L} \leftarrow \mathbf{I} - \mathbf{D}^{-\frac{1}{2}} S \mathbf{D}^{-\frac{1}{2}}$
  $\{C_x, C_o\} \leftarrow$ Spectral Clustering($\mathbf{L}$)
  Keep C_x where $P(\mathbf{Z}|\mathbf{Y}) \in C_x$
  Rank each $X_i \in C_x$ based on distance of $P(\mathbf{Z}|\mathbf{X}_i)$ to $P(\mathbf{Z}|\mathbf{Y})$
In case $\mathbf{Y}$ is not available, replace $P(\mathbf{Z}|\mathbf{Y})$ with $P(\mathbf{Z}|\mathbf{X}_{1:N})$.

We note that in case of irrelevant or malicious annotations, we assume that the corresponding signals will be moved to the private space and will not interfere with the time warping. Nevertheless, in order to ensure this, one can impose constraints on the warping process. This is easily done by modifying the DTW by imposing e.g., slope or global constraints such as the Itakura Parallelogram or the Sakoe-Chiba band, in order to constraint the warping path while also decreasing the complexity (c.f., Chap. 5, of [15]). Furthermore, other heuristics can be applied, e.g. firstly filter out the most irrelevant annotations by applying SG-DPCCA without time warping, or threshold the warping objective directly (Eq. 3.17).

3.6 Experiments

In order to evaluate our models, we present a set of experiments on both synthetic (Sec. 3.6.1) and real (Sec. 3.6.2 & 3.6.3) data.

3.6.1 Synthetic Data

For synthetic experiments, we employ a setting similar to [25]. A set of 2D spirals are generated as $\mathbf{X}_i = \mathbf{U}_i^T \tilde{\mathbf{Z}} \mathbf{M}_i^T + \mathbf{N}$, where $\tilde{\mathbf{Z}} \in \mathbb{R}^{2\times T}$ is the true latent signal which generates the $\mathbf{X}_i$, while the $\mathbf{U}_i \in \mathbb{R}^{2\times 2}$ and $\mathbf{M}_i \in \mathbb{R}^{T_i \times m}$ matrices impose random spatial and temporal warping. The signal is furthermore perturbed by additive noise via the matrix $\mathbf{N} \in \mathbb{R}^{2\times T}$. Each $\mathbf{N}(i,j) = e \times b$, where $e \sim \mathcal{N}(0,1)$ and b follows a Bernoulli distribution with $P(b=1) = 1$ for Gaussian and $P(b=1) = 0.4$ for spike noise. The length of the synthetic sequences varies, but is approximately 200.

This experiment can be interpreted as both of the problems we are examining. Viewed as a sequence alignment problem the goal is to recover the alignment of each noisy $\mathbf{X}_i$, where in this case the true alignment is known. Considering the problem of fusing multiple annotations, the latent signal $\tilde{\mathbf{Z}}$ represents the true annotation while the individual $\mathbf{X}_i$ form the set of noisy annotations containing annotation-specific characteristics. The goal is to recover the true latent signal (in DPCCA terms, $\mathbb{E}[\mathbf{Z}|\mathbf{X}_1, \ldots, \mathbf{X}_N]$).

The error metric we used computes the distance from the ground truth alignment ($\tilde{\mathbf{\Delta}}$) to the alignment recovered by each algorithm ($\mathbf{\Delta}$) [24], and is defined as:

$$\text{error} = \frac{\text{dist}(\mathbf{\Pi}, \tilde{\mathbf{\Pi}}) + \text{dist}(\tilde{\mathbf{\Pi}}, \mathbf{\Pi})}{T_\Delta + \tilde{T}_\Delta},$$

$$\text{dist}(\mathbf{\Pi}_1, \mathbf{\Pi}_2) = \sum_{i=1}^{T_\Delta^1} \min(\{||\pi_1^{(i)} - \pi_2^{(j)}||\})_{j=1}^{T_\Delta^2}), \tag{3.27}$$

where $\mathbf{\Pi}_i \in \mathbb{R}^{T_\Delta^i \times N}$ contains the indices corresponding to the binary selection matrices $\mathbf{\Delta}_i$, as defined in Sec. 3.3.1 (and [24]), while $\pi^{(j)}$ refers to the j-th row of $\mathbf{\Pi}$. For qualitative evaluation, in Fig. 3.4, we present an example of applying (D)PCTW on 5 sequences. As can be seen, DPCTW is able to recover the true, de-noised, latent signal which generated the noisy observations (Fig. 3.4(e)), while also aligning the noisy sequences (Fig. 3.4(c)). Due to the temporal modelling of DPCTW, the recovered latent space is almost identical to the true signal $\tilde{\mathbf{Z}}$ (Fig. 3.4(b)). PCTW on the other hand is unable to entirely remove the noise (Fig. 3.4(d)). Fig. 3.5 shows further results comparing related methods. CTW and GTW perform comparably for two sequences, both outperforming DTW. In general, PCTW seems to perform better than CTW, while DPCTW provides better alignment than other methods compared.

3.6.2 Real Data I: Fusing Multiple Annotations

In order to evaluate (D)PCTW in case of real data, we employ the SEMAINE database [12]. The database contains a set of audio-visual recordings of subjects interacting with operators. Each operator assumes a certain personality - happy, gloomy, angry and pragmatic - with a goal of inducing spontaneous emotions by the subject during a naturalistic

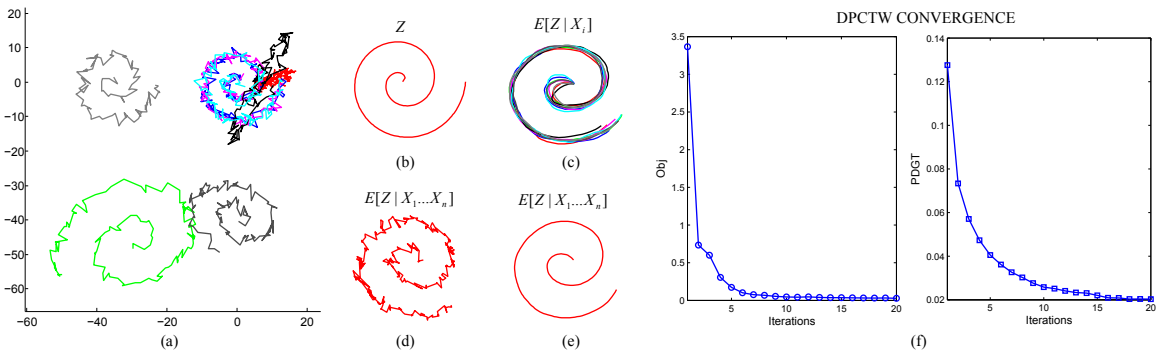

Figure 3.4: Noisy synthetic experiment. (a) Initial, noisy time series. (b) True latent signal from which the noisy, transformed spirals where attained in (a). (c) The alignment achieved by DPCTW. The shared latent space recovered by (d) PCTW and (e) DPCTW. (f) Convergence of DPCTW in terms of the objective (Obj) (Eq. 3.17) and the path difference between the estimated alignment and the true alignment path (PDGT).

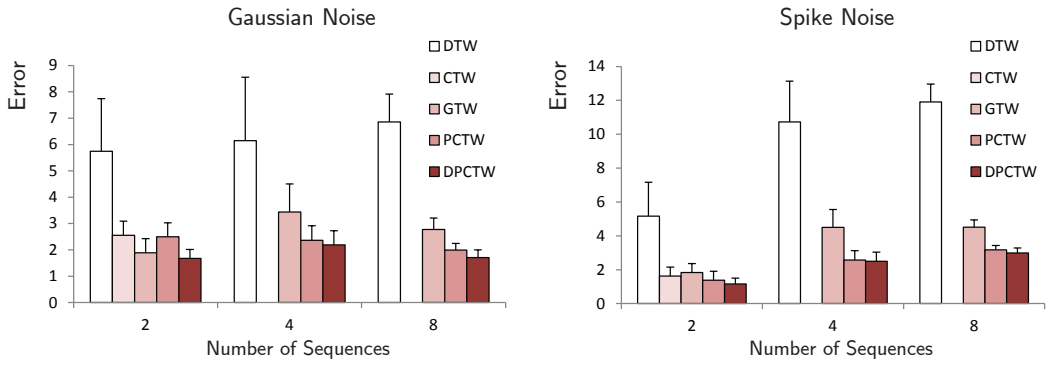

Figure 3.5: Synthetic experiment comparing the alignment attained by DTW, CTW, GTW, PCTW and DPCTW on spirals with spiked and Gaussian noise.

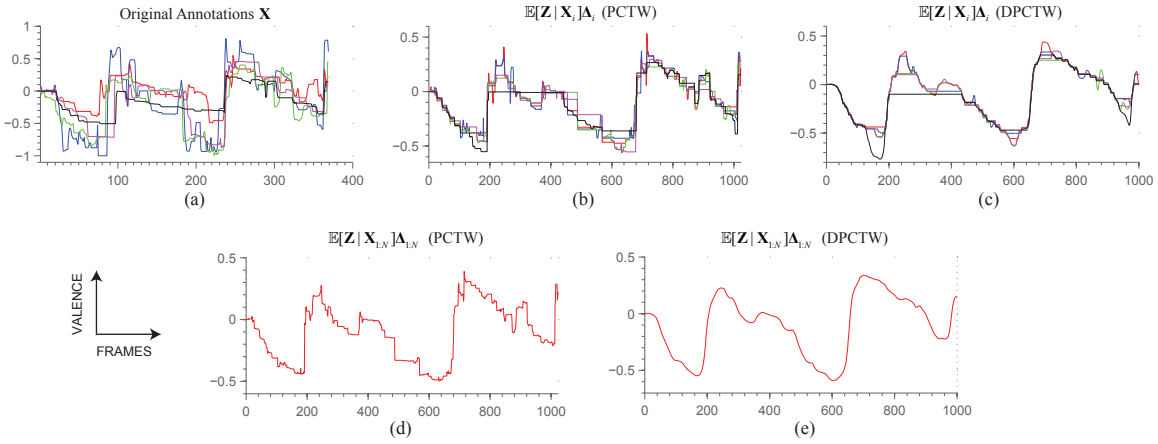

Figure 3.6: Applying (D)PCTW to continuous emotion annotations. (a) Original valence annotations from 5 experts. (b,c) Alignment obtained by PCTW and DPCTW respectively, (d,e) Shared space obtained by PCTW and DPCTW respectively, which can be considered as the "derived ground truth".

conversation. We use a portion of the database containing recordings of 6 different subjects, from over 40 different recording sessions, with a maximum length of 6000 frames per segment. As the database was annotated in terms of emotion dimensions by a set of experts (varying from 2 to 8), no single ground truth is provided along with the recordings. Thus, by considering $\mathbf{X}$ to be the set of annotations and applying (D)PCTW, we obtain $\mathbb{E}[\mathbf{Z}|\mathcal{D}] \in \mathbb{R}^{1\times T}$ (given all *warped* annotations)[9], which represents the shared latent space with annotator-specific factors and noise removed. We assume that $\mathbb{E}[\mathbf{Z}|\mathcal{D}]$ represents the ground truth. An example of this procedure for (D)PCTW can be found in Fig. 3.6. As can be seen, DPCTW provides a smooth, aligned estimate, eliminating temporal discrepancies, spike-noise and annotator bias. In this experiment, we evaluate our models on four emotion dimensions: valence, arousal, power, and anticipation (expectation).

To obtain features for evaluating the ground truth, we track the facial expressions of each subject via a particle filtering tracking scheme [14]. The tracked points include the corners of the eyebrows (4 points), the eyes (8 points), the nose (3 points), the mouth (4 points) and the chin (1 point), resulting in 20 2D points for each frame.

For evaluation, we consider a training sequence $\mathbf{X}$, for which the set of annotations $\mathcal{A}_x = \{\mathbf{a}_1, \ldots, \mathbf{a}_R\}$ is known. From this set ($\mathcal{A}_x$), we derive the ground truth $\mathcal{GT}_{\mathbf{X}}$ - for (D)PCTW, $\mathcal{GT}_{\mathbf{X}} = \mathbb{E}[\mathbf{Z}|\mathcal{A}_x]$. Using the tracked points $\mathcal{P}_{\mathbf{X}}$ for the sequence, we train a regressor to learn the function $f_x : \mathcal{P}_{\mathbf{X}} \rightarrow \mathcal{GT}_{\mathbf{X}}$. In (D)PCTW, $\mathcal{P}_x$ is firstly aligned with $\mathcal{GT}_x$ as they are not necessarily of equal length. Subsequently given a testing sequence $\mathbf{Y}$ with tracked points $\mathcal{P}_y$, using f_x we predict each emotion dimension ($f_x(\mathcal{P}_y)$). The

[9]We note that latent (D)PCTW posteriors used, e.g. $\mathbf{Z}|\mathbf{X}_i$ are obtained on time-warped observations, e.g. $\mathbf{Z}|\mathbf{X}_i\boldsymbol{\Delta}_\mathbf{i}$ (See Alg. 1)

procedure for deriving the ground truth is then applied on the annotations of sequence $\mathbf{Y}$, and the resulting $\mathcal{GT}_y$ is evaluated against $f_x(\mathcal{P}_y)$. The correlation coefficient of the $\mathcal{GT}_y$ and $f_x(\mathcal{P}_y)$ (after the two signals are temporally aligned) is then used as the evaluation metric for *all* compared methods.

The reasoning behind this experiment is that the "best" estimation of the ground truth (i.e. the gold standard) should maximally correlate with the corresponding input features - thus enabling any regressor to learn the mapping function more accurately.

We also perform experiments with the supervised variants of DPCTW, i.e. SG-DPCTW and SD-DPCTW. In this case, a set of features $\mathbf{Y}$ is used for inferring the ground truth, $\mathbf{Z}|\mathcal{D}$. Since we already used the facial trackings for evaluation, in order to avoid biasing our results[10], we use features from the audio domain. In particular, we extract a set of audio features consisting of 6 mel-frequency Cepstrum Coefficients (MFCC), 6 MFCC-Delta coefficients along with prosody features (signal energy, root mean squared energy and pitch), resulting in a 15 dimensional feature vector. The audio features are used to derive the ground truth with our supervised models, exactly acting an objective reference to our sequence. In this way, we impose a further constraint on the latent space: it should also explain the audio cues and not only the annotations, given that the two sets are correlated. Subsequently, the procedure described above for unsupervised evaluation with facial trackings is employed.

For regression, we employ RVM [19] with a Gaussian kernel. We perform both session-dependent experiments, where the validation was performed on each session separately, and session-independent experiments where different sessions were used for training/testing. In this way, we validate the derived ground truth generalisation ability (i) when the set of annotators is the same and (ii) when the set of annotators may differ.

Session-dependent and session-independent results are presented in Tables 3.1 and 3.2. We firstly discuss the unsupervised methods. As can be seen, taking a simple annotator average (A-AVG) gives the worse results (as expected), with a very high standard deviation and weak correlation. The model of Raykar et al. [16] provides better results, which can be justified by the variance estimation for each annotator. Modelling annotator bias and noise with (D)PCCA further improves the results. It is important to note that incorporating alignment is significant for deriving the ground truth; this is reasonable since when the annotations are misaligned, shared information may be modelled as individual factors or vice-versa. Thus, PCTW improves the results further while DPCTW provides the best results, confirming our assumption that combining dynamics, temporal alignment, modelling noise and individual-annotator bias leads to a more objective ground truth. Finally, regarding supervised models SG-DPCTW and SD-DPCTW, we can observe that the inclusion of audio features in the ground truth generation improves the results, with SG-DPCTW providing better correlated results than SD-DPCTW. This

[10]Since we use the facial points for *evaluating* the derived ground truth, if we had also used them for *deriving* the ground truth we would bias the evaluation procedure.

is reasonable since in SG-DPCTW the features $\mathbf{Y}$ are explicitly generated from the shared space, thus imposing a form of strict supervision, in comparison to SD-DPCTW where the inputs essentially elicit the shared space.

Table 3.1: Comparison of ground truth evaluation based on the correlation coefficient (COR), on session dependent experiments. The standard deviation over all results is denoted by σ.

	SD-DPCTW		**SG-DPCTW**		**DPCTW**		**PCTW**	
	COR	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$
Valence	**0.78**	*0.18*	**0.78**	*0.17*	**0.77**	*0.18*	0.70	*0.18*
Arousal	0.75	*0.18*	**0.77**	*0.19*	**0.75**	*0.22*	0.64	*0.22*
Power	0.78	*0.13*	**0.85**	*0.10*	0.77	*0.16*	0.76	*0.10*
Expectation	0.82	*0.09*	**0.83**	*0.10*	0.78	*0.11*	0.75	*0.16*
	DPCCA		**PCCA**		**RAYKAR** [16]		**A-AVG**	
	COR	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$
Valence	0.64	*0.21*	0.63	*0.20*	0.61	*0.20*	0.54	*0.36*
Arousal	0.63	*0.23*	0.63	*0.26*	0.60	*0.25*	0.42	*0.41*
Power	0.68	*0.16*	0.67	*0.18*	0.62	*0.22*	0.42	*0.36*
Expectation	0.68	*0.16*	0.74	*0.17*	0.62	*0.20*	0.48	*0.40*

Table 3.2: Comparison of ground truth evaluation based on the correlation coefficient (COR), on session independent experiments. The standard deviation over all results is denoted by σ.

	SD-DPCTW		**SG-DPCTW**		**DPCTW**		**PCTW**	
	COR	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$
Valence	**0.73**	*0.19*	**0.73**	*0.19*	**0.72**	*0.22*	0.66	*0.24*
Arousal	**0.74**	*0.15*	**0.74**	*0.17*	**0.71**	*0.20*	0.61	*0.23*
Power	0.72	*0.28*	**0.75**	*0.24*	**0.72**	*0.34*	0.70	*0.19*
Expectation	0.76	*0.21*	**0.76**	*0.15*	**0.73**	*0.20*	0.70	*0.18*
	DPCCA		**PCCA**		**RAYKAR** [16]		**A-AVG**	
	COR	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$	**COR**	$\boldsymbol{\sigma}$
Valence	0.62	*0.28*	0.58	*0.23*	0.57	*0.27*	0.53	*0.33*
Arousal	0.59	*0.23*	0.52	*0.28*	0.50	*0.29*	0.33	*0.40*
Power	0.60	*0.26*	0.58	*0.27*	0.57	*0.27*	*0.39*	*0.31*
Expectation	0.63	*0.20*	0.64	*0.25*	0.63	*0.22*	0.44	*0.39*

Ranking Annotations

We perform the ranking of annotations as proposed in Algorithm 2 to a set of emotion dimension annotations from the SEMAINE database.

In Fig. 3.7(a), we illustrate an example where an irrelevant structured annotation (sinusoid), has been added to a set of five true annotations. Obviously the sinusoid can be considered a spammer annotation since essentially, it is independent of the actual sequence at hand. In the figure we can see that (i) the derived ground truth is not affected by the spammer annotation, (ii) the spammer annotation is completely captured in the private space, and (iii) that the spammer annotation is detected in the distance matrix of $\mathbb{E}[\mathbf{Z}|\mathbf{X}_i]$ and $\mathbb{E}[\mathbf{Z}|\mathbf{X}]$.

In Fig. 3.7(b), we present an example where a set of 5 annotations has been used along with 8 spammers. The spammers consist of random Gaussian distributions along with structured periodical signals (i.e. sinusoids). We can see that it is difficult to discriminate the spammers by analysing the distance matrix of $\mathbf{X}$ since they do maintain some correlation with the true annotations. By applying Algorithm 2, we obtain the distance matrix of the latent posteriors $\mathbf{Z}|\mathbf{X}_i$ and $\mathbf{Z}|\mathcal{D}$. In this case, we can clearly detect the cluster of annotators which we should keep. By applying spectral clustering, the spammer annotations are isolated in a single cluster, while the shared space along with the true annotations fall into the other cluster. This is also obvious by observing the inferred weight vector ($\mathbf{W}$), which is near-zero for sequences 6-14, implying that the shared signal is ignored when reconstructing the specific annotation (i.e. the reconstruction is entirely from the private space). Finally, this is also obvious by calculating the KL divergence comparing each individual posterior $\mathbf{Z}|X_i$ to the shared space posterior given all annotations $\mathbf{Z}|\mathcal{D}$, where sequences 6-14 have a high distance while 1-5 have a distance which is very close to zero.

In Fig. 3.7(c), we present another example where in this case, we joined two sets of annotations which were recorded for two distinct sequences (annotators 1-6 for sequence A and annotators 7-12 for sequence B). In the distance matrix taken on the observations $\mathbf{X}$, we can see how the two clusters of annotators are already discriminable, with the second cluster, consisting of annotations for sequence B, appearing more correlated. We use the facial trackings for sequence A (tracked as described in this section) as the features $\mathbf{Y}$, and then apply Algorithm 2. As can be seen in the distance matrix of $[\mathbf{Z}|\mathbf{X}_i, \mathbf{Z}|\mathbf{Y}]$, (i) the two clusters of annotators have been clearly separated, and (ii) the posterior of features $\mathbf{Z}|\mathbf{Y}$ clearly is much closer to annotations 1-6, which are the true annotations of sequence A.

3.6.3 Real Data II: Action Unit Alignment

In this experiment we aim to evaluate the performance of (D)PCTW for the temporal alignment of facial expressions. Such applications can be useful for methods which require pre-aligned data, e.g. AAM (Active Appearance Models). For this experiment, we use a portion of the MMI database which contains more than 300 videos, ranging from 100 to 200 frames. Each video is annotated (per frame) in terms of the temporal phases of each Action Unit (AU) manifested by the subject being recorded, namely neutral, onset, apex and offset. For this experiment, we track the facial expressions of each subject capturing 20 2D points, as in Sec. 3.6.2.

Given a set of videos where the same AU is activated by the subjects, the goal is to temporally align the phases of each AU activation across *all* videos containing that AU, where the facial points are used as features. In the context of DPCTW, each $\mathbf{X}_i$ is the facial points of video i containing the same AU, while $\mathbf{Z}|\mathbf{X}_i$ is now the common latent space given video i, the size of which is determined by cross-validation, and is constant over all experiments for a specific noise level.

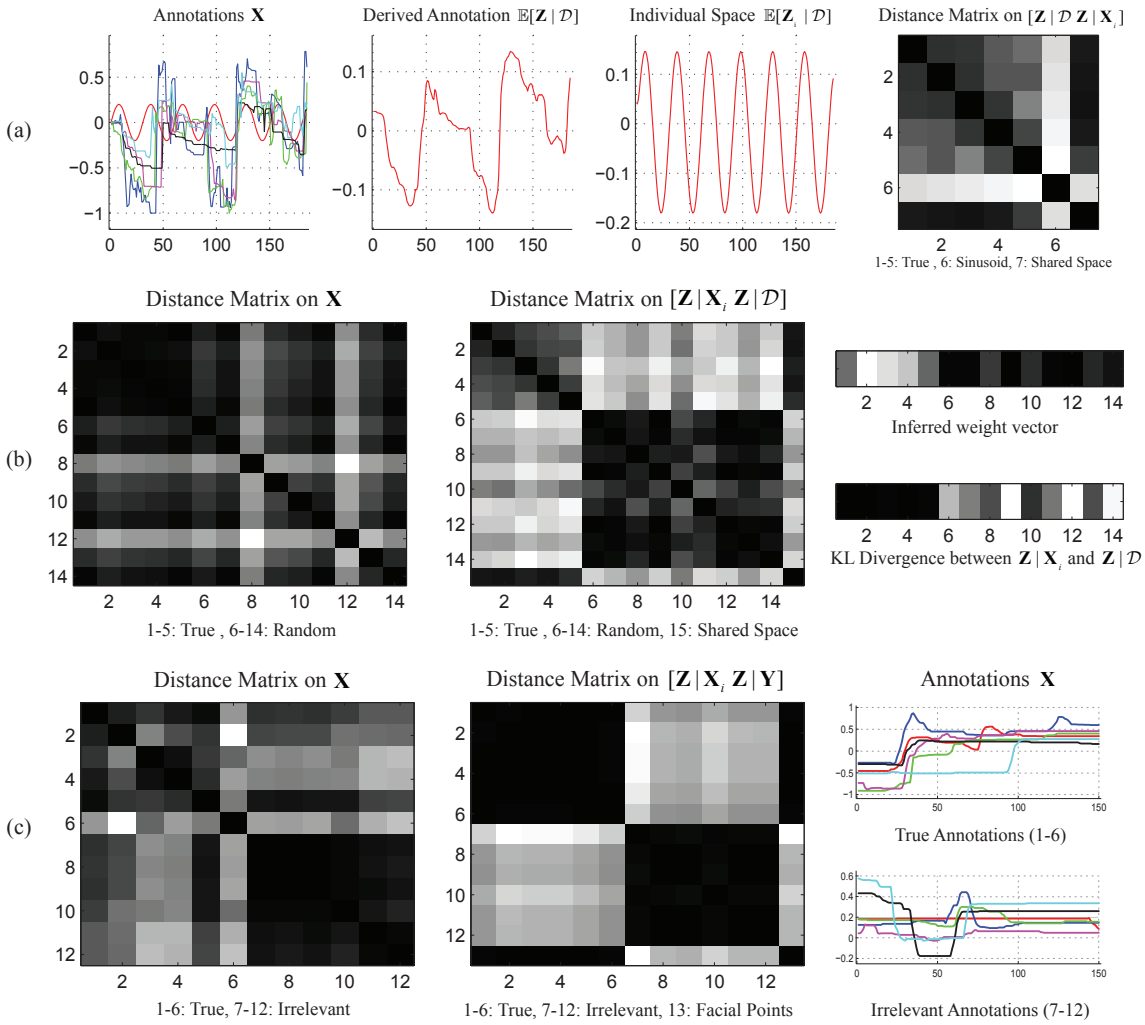

Figure 3.7: Annotation filtering and ranking (black - low, white - high). (a) Experiment with a structured false annotation (sinusoid). The shared space is not affected by the false annotation, which is isolated in the individual space. (b) Experiment with 5 true and 9 spammer (random) annotations. (c) Experiment with 6 true annotations, 7 irrelevant but correlated annotations (belonging to a different sequence). The facial points $\mathbf{Y}$, corresponding to the 6 true annotations, were used for supervision (with SG-DPCCA).

In Fig. 3.8 we present results based on the number of misaligned frames for AU alignment, on all action unit temporal phases (neutral, onset, apex, offset) for AU 12 (smile), on a set of 50 pairs of videos from MMI. For this experiment, we used the facial features relating to the lower face, which consist of 11 2D points. The features were perturbed with sparse spike noise in order to simulate the mis-detection of points with detection-based trackers, in order to evaluate the robustness of our techniques. Values were drawn from the normal distribution $\mathcal{N}(0,1)$ and added (uniformly) to 5% of the length of each video. We gradually increased the number of features perturbed by noise from 0 to 4. To evaluate the accuracy of each algorithm, we use a robust, normalised metric. In more detail, let us say that we have two videos, with features $\mathbf{X}_1$ and $\mathbf{X}_2$, and AU annotations $\mathcal{A}_1$ and $\mathcal{A}_2$. Based on the features, the algorithm at hand recovers the alignment matrices Δ_1 and Δ_2. By applying the alignment matrices on the AU annotations ($\mathcal{A}_1\Delta_1$ and $\mathcal{A}_2\Delta_2$), we know to which temporal phase of the AU each aligned frame of each video corresponds to. Therefore, for a given temporal phase (e.g., neutral), we have a set of frame indices which are assigned to the specific temporal phase in video 1, Ph_1 and video 2, Ph_2. The accuracy is then estimated as $\frac{Ph_1 \cap Ph_2}{Ph_1 \cup Ph_2}$. This essentially corresponds to the ratio of correctly aligned frames to the total duration of the temporal phase accross the aligned videos.

As can be seen in the average results in Fig. 3.8, the best performance is clearly obtained by DPCTW. It is also interesting to highlight the accuracy of DPCTW on detecting the apex, which essentially is the peak of the expression. This can be attributed to the modelling of dynamics, not only in the shared latent space of all facial point sequences but also in the domain of the individual characteristics of each sequence (in this case identifying and removing the added temporal spiked noise). PCTW peforms better on average compared than CTW and GTW, while the latter two methods perform similarly. It is interesting to note that GTW seems to overpeform CTW and PCTW for aligning the apex of the expression for higher noise levels. Furthermore, we point-out that the Gauss-Newton warping used in GTW is likely to perform better for longer sequences. Example frames from videos showing the unaligned and DPCTW-aligned videos are shown in Fig. 3.9 .

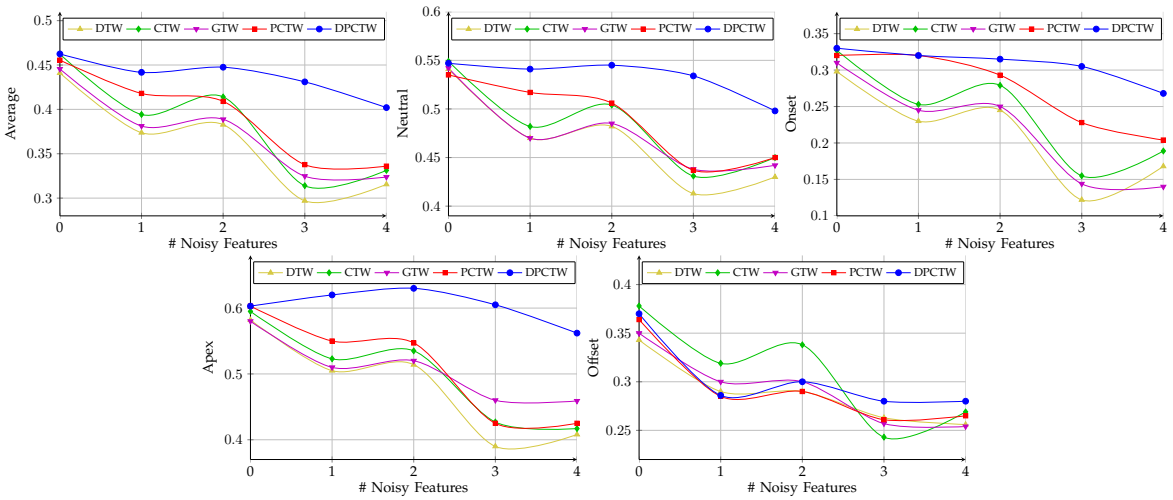

Figure 3.8: Accuracy of DTW, CTW, GTW, PCTW and DPCTW on the problem of action unit alignment under spiked noise added to an increasing number of features for AU = 12 (smile).

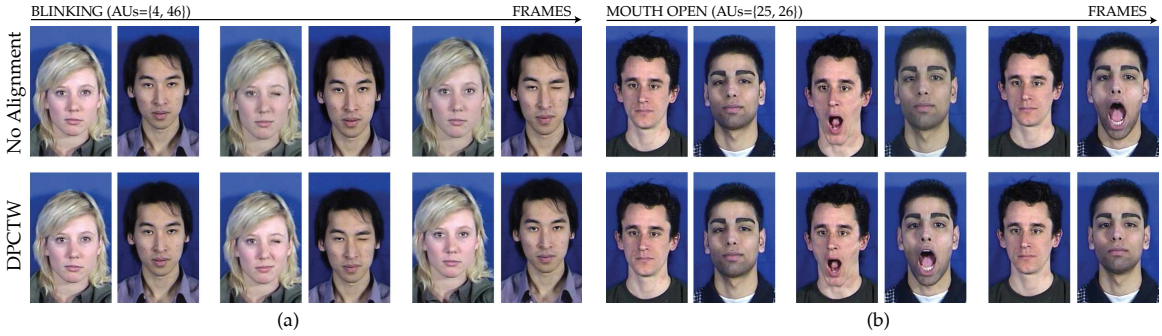

Figure 3.9: Example stills from a set of videos from the MMI database, comparing the original videos to the aligned videos obtained via DPCTW under spiked noise on 4 2D points. (a) Blinking, AUs 4 and 46. (b) Mouth open, AUs 25 and 27.

Chapter 4

FROG Interest Model

In order to deal with the problem of interest prediction under realistic and uncontrolled scenarios such as in FROG, we firstly discretize the continuous annotations to cover three classes, which are of main interest. In more detail, we solve a series of binary classification problems, as seen in Fig. 4.1. As features, we utilise the pose and 3D points, as generated by the FROG tracker. The first step involves deciding whether the input points form a valid facial structure, a facial shape in other words. Secondly, given that we have a shape structure, we infer whether the tracked person is interested or not. Finally, given that the person is interested, we derive the level of interest, be it low or high.

Regarding the data used for training the models, we have a total of approximately 65000 frames at 64 FPS, where approximately 30000 frames are assigned to low interest and 30000 frames to high interest, while the rest are assigned to no interest (3000 frames) and the class where no face is recognised (tracker error, 2000 frames). The imbalanced nature of some classes is naturalistic to the data at hand, and has been dealt with by selecting the proper misclassification penalty within SVM. During cross-validation, the accuracy of level 0 (Face Verification) was 99.9%, of Level 1 (Interest vs. No-Interest) 99.8% and finally, Level 2 (Low vs. High Interest) 97.5%. The confusion matrices when training on the entire set utilising the parameters inferred by cross-validation, are shown in Table 4.1.

LEVEL 0		LEVEL 1		LEVEL 2	
0.999	0.001	0.992	0.008	0.993	0.007
0.000	1.000	0.000	1.000	0.018	0.982

Table 4.1: Confusion matrices for Level 0 (Face Verification), Level 1 (No Interest vs. Interest) and level 2 (Low vs. High Interest).

We note that for each step, binary Support Vector Machine (SVM) classifiers were employed. As aforementioned, we utilised the pose and the 3D landmarks, resulting in 201 feature vector dimensionality. For SVM, we used a Radial Basis Function (RBF) kernel, i.e. for input $\mathbf{x_i}$, $K(\mathbf{x}, \mathbf{x}_i) = \exp\left\{\frac{-(\mathbf{x}-\mathbf{x}_i)^2}{\mathbf{r}^2}\right\}$, with $\mathbf{r}^2$ being the length scale. As

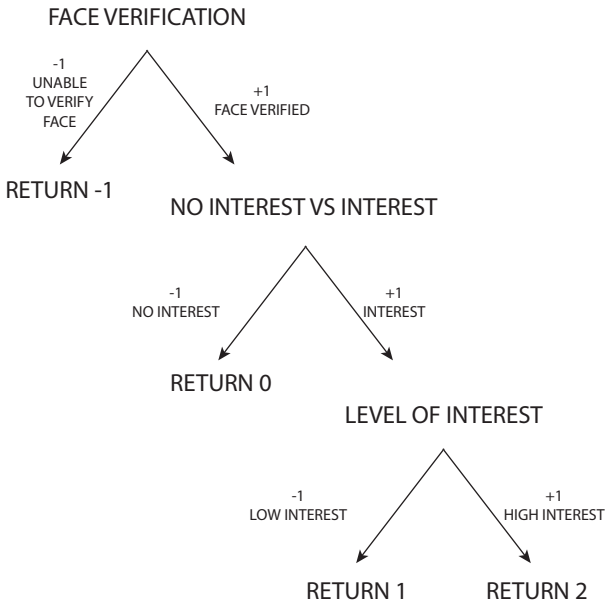

Figure 4.1: Interest prediction model for FROG.

it is well known, SVM's, for a given class label y_i, solve the following problem:

$$\min \frac{1}{2}||\mathbf{w}||^2 s.t. y_i(\mathbf{x}_i\mathbf{w} + b) - 1 \geq 0 \forall i \tag{4.1}$$

where $\mathbf{w}$ is the inferred set of weights, and y_i the class labels, where y_i= $+1$ or -1, and b simply represents a constant bias. Examples of using the interest prediction model on FROG data are depicted in Figure 4.2.

Figure 4.2: Sample output by utilising the Interest Prediction model for FROG.

Chapter 5

Conclusion

A method for prediction of the visitors' implicit affective feedback (e.g. attention and interest) was developed for FROG project. The main goal was to develop a set of visual methods for detecting human affective states including users' positive and negative reactions to FROG robot and their overall level of interest and engagement in the current interaction with FROG. Facial landmarks as well as face pose were used for building our models. Based on our experiments in both synthetic and real FROG data, this method meets all the requirements of the FROG project regarding the prediction of the visitors' implicit affective feedback.

Bibliography

[1] R. J. Aumann. Agreeing to disagree. *The Annals of Statistics*, 4(6):pp. 1236–1239, 1976.

[2] F. R. Bach and M. I. Jordan. A Probabilistic Interpretation of Canonical Correlation Analysis. Technical report, 2006.

[3] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[4] M. W. Browne. The maximum-likelihood solution in inter-battery factor analysis. *British Journal of Mathematical and Statistical Psychology*, 32(1):75–86, 1979.

[5] Z. Ghahramani, M. I. Jordan, and P. Smyth. Factorial hidden markov models. In *Machine Learning*. MIT Press, 1997.

[6] Z. Ghahramani and S. T. Roweis. Learning nonlinear dynamical systems using an em algorithm. In *Advances in Neural Information Processing Systems 11*, pages 599–605. MIT Press, 1999.

[7] H. Gunes, B. Schuller, M. Pantic, and R. Cowie. Emotion representation, analysis and synthesis in continuous space: A survey. In *IEEE Int. Conf. on Aut. Face and Gesture Recognition 2011, EmoSPACE WS*, pages 827–834, Santa Barbara, USA, 2011.

[8] M. A. Hasan. On multi-set canonical correlation analysis. In *Proc. of the Int. Joint Conf. on Neural Networks*, IJCNN'09, pages 2640–2645, Piscataway, NJ, USA, 2009. IEEE Press.

[9] M. Kim and V. Pavlovic. Discriminative Learning for Dynamic State Prediction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(10):1847–1861, 2009.

[10] A. Klami and S. Kaski. Probabilistic approach to detecting dependencies between data sets. *Neurocomput.*, 72(1-3):39•46, Dec. 2008.

[11] M. A. Larkin, G. Blackshields, N. Brown, R. Chenna, P. McGettigan, et al. Clustal w and clustal x version 2.0. *Bioinformatics*, 23(21):2947–2948, 2007.

[12] G. McKeown, M. F. Valstar, R. Cowie, and M. Pantic. The semaine corpus of emotionally coloured character interactions. In *2010 IEEE Int. Conf. on Multim. and Expo*, pages 1079 –1084, 2010.

[13] M. A. Nicolaou, H. Gunes, and M. Pantic. Continuous prediction of spontaneous affect from multiple cues and modalities in valence-arousal space. *IEEE Transactions on Affective Computing*, pages 92–105, 2011.

[14] I. Patras and M. Pantic. Particle filtering with factorized likelihoods for tracking facial features. In *Proc. of IEEE Int. Conf. on Automatic Face and Gesture Recognition*, pages 97–102, 2004.

[15] L. Rabiner and B. H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall, united states ed edition, Apr. 1993.

[16] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy. Learning from crowds. *Journal of Machine Learning Research*, 11:1297–1322, 2010.

[17] S. Roweis and Z. Ghahramani. A unifying review of linear gaussian models. *Neural Comput.*, 11:305–345, 1999.

[18] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, Aug. 2000.

[19] M. E. Tipping. Sparse Bayesian Learning and the Relevance Vector Machine. *Journal of Machine Learning Research*, 1:211–244, 2001.

[20] L. R. Tucker. An inter-battery method of factor analysis. *Psychometrika*, 23(2):111–136, 1958.

[21] R. Van der Merwe and E. Wan. The square-root unscented kalman filter for state and parameter-estimation. In *In Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, 2001.*, volume 6, pages 3461 –3464 vol.6, 2001.

[22] S. Yu, K. Yu, V. Tresp, H.-P. Kriegel, and M. Wu. Supervised probabilistic principal component analysis. In *Proc. of the 12th ACM SIGKDD Int. Conf. on Knowledge discovery and data mining*, KDD '06, pages 464–473, New York, NY, USA, 2006. ACM.

[23] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang. A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(1):39–58, 2009.

[24] F. Zhou and F. De la Torre Frade. Generalized time warping for multi-modal alignment of human motion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012.

[25] F. Zhou and F. D. la Torre. Canonical time warping for alignment of human behavior. In *Advances in Neural Information Processing Systems 22*, pages 2286–2294, 2009.