Feature Reduction for Dimensional Emotion Recognition in Human-Robot Interaction

Ntombikayise Banda
Computer Laboratory
University of Cambridge, UK
Email: nb395@cam.ac.uk

Andries Engelbrecht
Department of Computer Science
University of Pretoria, ZA
Email: engel@cs.up.ac.za

Peter Robinson
Computer Laboratory
University of Cambridge, UK
Email: pr10@cam.ac.uk

***Abstract*—The introduction of social robots in human living spaces has brought to attention the need for robots to be equipped with emotion recognition capabilities to facilitate natural and social human-robot interactions. This paper explores the recognition of continuous dimensional emotion from facial expressions. It further investigates the use of principal component analysis (PCA), locality preserving projections (LPP) and factor analysis (FA) for reduction of the many features that are typically produced by facial feature extraction algorithms. The reduced features sets are modelled using Nonlinear AutoRegressive with eXogenous inputs Recurrent Neural Networks (NARX-RNN). The results show that PCA significantly outperfoms both LPP and FA techniques, and that the NARX-RNN model is a powerful predictor of continuous emotion.**

I. Introduction

Robots have become prevalent in our society with their use spanning many domains such as manufacturing, education and health care. Those applied in social environments are autonomous and are typically endowed with cognitive abilities to emulate human reasoning, experiential learning and communication. Their interaction with humans is a key element that necessitates in-depth research and careful design. Reeves and Nass [1] argue that people tend to treat computers (and other media) as if they were real people and their interactions are fundamentally social and natural. Since emotions are central to human experience and behavior and inherent in all forms of communication [2], it is important for social cognitive robots to be equipped with an affective component that can recognize emotion and respond appropriately. This will allow for natural interactions between humans and robots.

Ekman and Friesen [3] suggest that non-verbal behaviours are the primary vehicles for expressing emotion. This is corroborated by Mehrabian [4] who found that the predominant form of communication is non-verbal with the body language and tone of voice accounting for 55% and 38% of affective information respectively, and spoken words only accounting for the remaining 7%. Thus, the analysis of non-verbal communication is a key component in the recognition and synthesis of emotion in robots.

According to research in psychology, there are three theoretical perspectives on emotion: categorical, dimensional and appraisal-based theories [5]. The categorical emotion theory is based on the findings that there exists six basic, universally-recognised emotions with prototypical facial expressions namely, anger, disgust, fear, joy, sadness and surprise [6]. Baron-Cohen *et al.* [7] argued that the list is not reflective of the emotions typically experienced by people in work and social environments and as a result incorporated cognitive mental states such as frustration and interest. In contrast, the dimensional emotion theory argues that emotions are not independent but are related in a systematic way and can be represented on a common multidimensional space [8], [9]. The two primary dimensions are *valence* which refers to how positive or negative an emotion is, and *arousal* which describes the intensity of an emotion (ranging from sleepiness to excitement). The dimensional emotion model therefore encompasses discrete emotion classes and provides a wider range of emotion varieties as shown in Figure 1. This emotion model will allow a robot to model variability in emotion and be able to distinguish intensities of emotion, for example anger, which ranges from mild irritation to intense fury. The appraisal-based emotion theory claims that emotions arise from one's perceptions and cognitive evaluations of their circumstances [10]. The theory accounts for individual variances of emotional reactions to the same event. However, its application to automatic emotion recognition is still in the early stages.

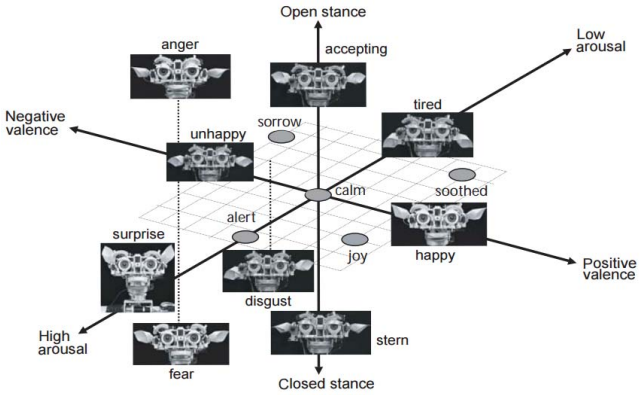

Fig. 1. This figure shows various discrete emotions expressed by the robot Kismet that fall along the valence (pleasant or unpleasant), arousal (high or low) and stance (advance or withdraw) dimensions. [11]

978-1-4799-7560-0/15 $31.00 © 2015 IEEE
DOI 10.1109/SSCI.2015.119

This paper focuses on the recognition of dimensional emotion from facial expressions. It further addresses the challenge of processing high-dimensional emotion features. It is not uncommon for facial feature extraction techniques to produce features in excess of 1000 dimensions for facial expression analysis. Recently, the Audio-Visual Emotion Challenge (AVEC) - a workshop aimed at providing a platform for comparative analysis of emotion prediction approaches using the same dataset - provided participants with baseline feature sets consisting of 1188 video features and 1841 audio features [12]. Various studies emanating from the AVEC competition used these features or added to them resulting in increased feature sets [13], [14]. The cost of using high-dimensional features is often evident in the poor performance of models or increased model complexities. In the interest of building efficient and compact emotion recognition systems that will allow for real-time processing of emotion in human-robot interactions, it is imperative to consider feature reduction that will yield much lower features. This study provides a comparative analysis of feature reduction techniques for continuous dimensional emotion recognition. Although such analyses exist in other domains and in emotion recognition, they have mostly been applied for categorical emotions which is a classification task rather than the regression task that is at hand. The techniques compared in this work are principal component analysis (PCA), factor analysis (FA) and locality preserving projections (LPP). These techniques are described in section III and the results thereof are presented in section IV-C.

II. Emotion Recognition

This section describes the affective computing framework (depicted in Figure 2) for analysis of facial expressions to inform a robot of a person's cognitive state. A robot is typically fitted with a camera to capture its environment. The image sequences captured by the camera are fed into a face tracker which attempts to locate a face on each image. Once located, the facial expression features are extracted from the image sequence and a dimension reduction technique is applied prior to modelling the data.

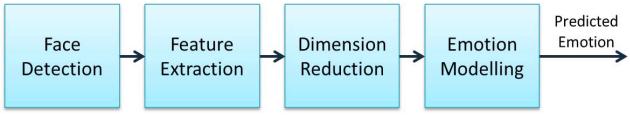

Fig. 2. Emotion Recognition Framework

The training task is of a supervised form where target emotion labels are fed into the model to learn mappings between the input data and desired emotion. The subsections below provide detailed information of the analysis of emotion.

A. Face Tracking

Tracking of the face is achieved through the use of the GAVAM-CLM tracker [15] which combines non-rigid face tracking and rigid head pose tracking approaches for accurate location of the face. Non-rigid tracking approach refers to locating facial landmarks of interest from an image such as the corner of the eyes and the outline of the lips. Rigid head pose tracking refers to the estimation of the location and orientation of the head. The tracked face is thereafter cropped to remove environmental background which could present itself as noise to the emotion recognition model. The cropped faces are thereafter normalized to compensate for orientation and illumination variations.

B. Feature Extraction

The cropped image sequences from the previous subsection are passed onto a temporal local binary pattern algorithm to extract facial features that will enable sucessful modelling of the face expressions. The temporal local binary algorithm used in this work is an extension of the original local binary pattern (LBP) operator [16] which captures the motion and appearance of an image sequence and produces a feature descriptor that describes the dynamic textures.

The original LBP operator assigns a code to every pixel p of an image by thresholding the neighbourhood of the pixel (that is, its 8 immediate neighbours) and assigning the value 1 if the grayscale value of a neighbouring pixel value is greater than that of the center pixel, and 0 otherwise. The threshold result is considered as an 8-bit code which is converted to a decimal value for convenience. This is represented by

$$LBP_p = \sum_{n=0}^{N-1} s(g_n - g_p)2^n, \quad s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{1}$$

where n indexes the neighbouring pixels ($N = 8$), g_n is the greyscale value of a neighbouring pixel and g_p is the greyscale value of centre pixel p.

Once each pixel has been assigned a code, the image is sectioned into a 3×3 grid of non-overlapping blocks to capture micro-patterns, such as edges and flat areas, that could help discriminate between different facial expressions. A histogram of each block is computed and used as a texture descriptor. Uniform patterns are applied which reduce the dimensionality of the histogram from 256 ($2^N = 2^8$) to 59. The basic LBP operator captures the spatial domain information of an image. To incorporate temporal features, the local binary patterns of three orthogonal planes - XY, XT and YT - are computed and the statistical information of the three planes are concatenated into a single histogram. The XY-LBP accounts for the appearance statistics while the XT-LBP and YT-LBP encode the spatial-temporal co-occurrence statistics as shown in Figure 3. This extended algorithm is called LBP-TOP [17]. A complete feature vector is obtained by concatenating the histograms of each block over the three orthogonal planes resulting in a 1593-dimensional vector (59 features $\times$ 3 planes $\times$ 9 blocks).

C. Feature Reduction

Machine learning algorithms are known to degrade in performance when faced with many features that are not necessary for predicting the desired output - a concept known

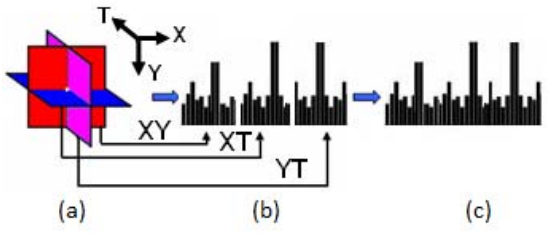

Fig. 3. (a) Three planes from which spatio-temporal local features are extracted (b) LBP histogram from each plane (c) Concatenated feature histogram. [17]

as the *curse of dimensionality*. Therefore the selection or extraction of relevant features lead to efficient modelling of data. The reader is referred to section III for a description of the feature reduction techniques investigated in this work. The reduced feature set serves as input to a machine learning model to learn facial expression patterns that can predict the correct emotion. The learning model applied is discussed in the following subsection.

D. Emotion Modeling: NARX Recurrent Neural Network

The modeling of emotion remains a challenging task due to the large variance in emotion expressions and the temporal nature of emotion, amongst other factors. This can be addressed by employing prediction models that capture the temporal dynamics of emotion as it unfolds. One such model is the Nonlinear AutoRegressive with eXogenous inputs Recurrent Neural Network (NARX-RNN) which is a dynamic network with feedback connections (as depicted in Figure 4) that allow the model to retain information about past inputs and to learn correlations between temporally distant events. It has shown remarkable success in tasks such as time series analysis [18], traffic modelling [19] and grammatical inference [20].

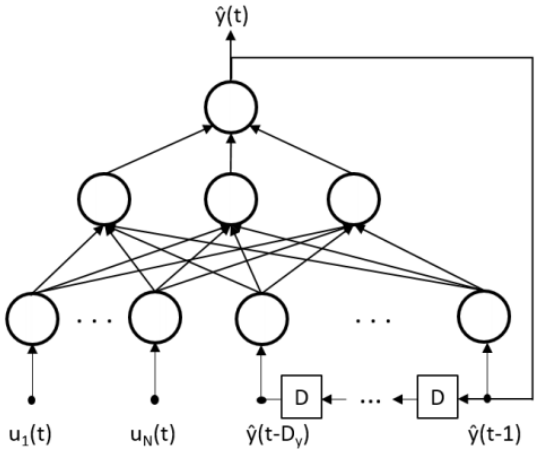

Fig. 4. Architecture of the NARX recurrent neural network

The NARX-RNN is characterized by the non-linear relations between past outputs, current and past independent (exogenous) inputs. The mathematical representation of the model is

$$y(t) = f(\mathbf{u}(t-D_u), \cdots, \mathbf{u}(t-1), \mathbf{u}(t), \\ y(t-D_y), \cdots, y(t-1)) \tag{2}$$

where $\mathbf{u}(t) \in \mathbb{R}^N$ and $y(t) \in \mathbb{R}^1$ are inputs and output of the network at time t, D_u and D_y are the input and output orders, and f is a non-linear function. For this work, past inputs are disregarded and D_u is set to zero. The function f is approximated by a Multilayer Perceptron (MLP) consisting of three layers, namely, the input, hidden and output layer with recurrent connections from the output to the input layer. The input to the MLP at time t becomes the concatenated vector of the exogenous features and past predicted outputs as shown:

$$\mathbf{x}(t) = [\mathbf{u}(t), y(t-D_y), \cdots, y(t-1)]. \tag{3}$$

The hidden layer computes a nonlinear transformation of the input with the sigmoid or hyperbolic tangent being the common choice for activation functions. A similar transformation is applied from the hidden layer to the output layer producing the predicted outputs. The hidden layer states $\mathbf{h}(t)$ and output prediction $\hat{y}(t)$ are computed as follows:

$$\mathbf{h}(t) = \Psi(\mathbf{W}_{hx} \cdot \mathbf{x}(t) + b_h) \tag{4}$$

$$\hat{y}(t) = \Gamma(\mathbf{W}_{yh} \cdot \mathbf{h}(t) + b_y) \tag{5}$$

where $\mathbf{W}_{hx}$ and $\mathbf{W}_{yh}$ are the input-to-hidden and hidden-to-output weight matrices respectively, b_h and b_y and the biases and Ψ and Γ are the activation functions. The network weights determine the significance of a node in the prediction of emotion. Ψ and Γ are both set to hyperbolic tangents in this work. The authors break away from the common practise of applying a linear activation function at the output layer by using a hyperbolic tangent function instead as it has the added effect of re-scaling the output predictions to lie between -1 and 1; a necessary step prior to feeding back the predictions as delayed output.

The optimal weights of the model $\theta = [\mathbf{W}_{hx}\ \mathbf{W}_{yh}\ b_h\ b_y]$ are learnt through the minimization of a loss (objective) function, the mean squared error (MSE), which measures the deviations of the predicted outputs, $\hat{\mathbf{y}}$, from the target outputs, $\mathbf{y}$. The loss function is defined as

$$J(\theta) = \frac{1}{2m} \sum_c \sum_t (y(t) - \hat{y}(t))^2 \tag{6}$$

where m is the total number of instances. A regularization term is added to the loss function,

$$J(\theta) = J(\theta) + \frac{\lambda}{2m} \sum_j \sum_k (w_{jk})^2, \tag{7}$$

to penalize large weights that could lead to the model overfitting the training data. The penalty parameter (λ) is usually determined through a cross-validation exercise. The weights are

adjusted concurrently through an iterative process that applies the backpropagation-through-time algorithm and the L-BFGS optimization algorithm. Details of the training procedure can be found in [21].

III. Dimensionality Reduction

Dimensionality reduction is an essential preprocessing technique for high-dimensional data as too many features could lead to overfitting of models and an increase in noise. There are two basic approaches to dimensionality reduction: feature selection and feature extraction. Feature selection is a process of selecting a subset of relevant features from the original feature set, while feature extraction creates a new feature set by transforming the existing features into a lower dimension. The data transformation can be linear or nonlinear.

This paper investigates the application of the following linear feature extraction techniques for analysis of temporal data: principal component analysis (PCA), factor analysis (FA) and locality preserving projections (LPP). These techniques have been shown to be effective in different problem domains, but have not been compared to each other in continuous emotion recognition tasks.

The subsections below provide an overview of each technique and list the steps involved in the data transformation. Each technique receives input data $\mathbf{x} \in \mathbb{R}^m$, and the goal is to find a linear transformation $\mathbf{A}$ that produces the reduced dataset $\mathbf{y} = \mathbf{A}'\mathbf{x} \in \mathbb{R}^d$ where $d < m$ and $\mathbf{A}'$ is the transposed matrix of $\mathbf{A}$.

A. Principal Component Analysis

Principal component analysis is a popular dimensionality reduction method. It operates on the basis that highly correlated features carry redundant information. It therefore reduces the number of variables by decorrelating input vectors.

The goal of PCA is to find a direction $\mathbf{A}$ that maximizes the variance of the projections of all input features $\mathbf{x}_j$ for dimension $j = 1, \cdots, N$. This is achieved through the following steps:

Step 1: Center the data

Subtract the mean for each data dimension to produce a dataset with a zero mean,

$$\mathbf{x}_j = \mathbf{x}_j - \bar{\mathbf{x}}_j \tag{8}$$

where $\bar{\mathbf{x}}_j$ is the average of all data points in dimension j.

Step 2: Calculate the covariance matrix of dataset $\mathbf{x}$

$$\mathbf{C}_x = \frac{1}{n}\mathbf{x}\mathbf{x}' \tag{9}$$

The matrix $\mathbf{C}_x$ captures the covariance between all pairs of features. The covariance values reflect the noise and redundancy in the features.

Step 3: Compute eigenvalues and eigenvectors

PCA works on the assumption that the transformed data, $\mathbf{y}$, should be as uncorrelated as possible. This is equivalent to the off-diagonal elements of the covariance matrix of the transformed data $\mathbf{C}_y$ (interactions between the variables) being as close to zero as possible. The covariance matrix $\mathbf{C}_y$ is approximated by

$$\mathbf{C}_y = \mathbf{A}\mathbf{C}_x\mathbf{A}' \tag{10}$$

where, given a square covariance matrix $\mathbf{C}_x$, the eigenvalue decomposition method computes the matrix $\mathbf{A}$ such that $\mathbf{C}_y$ is diagonal. The resulting $\mathbf{C}_y$ contains eigenvalues while matrix $\mathbf{A}$ contains eigenvectors.

The eigenvalues and eigenvectors are arranged with respect to the descending order of the eigenvalues (relaying the order of importance). The eigenvalues indicate how much of the data's variability is explained by its corresponding eigenvector, and the eigenvectors indicate the directions of the principal components which are orthogonal to each other. The first principal component therefore accounts for maximal amount of the total variance in the observed variables.

One can thereafter select the first d components that explain most of the variance in the data. This will yield a reduced eigenvector matrix.

Step 4: Apply linear mapping

The transformed data is therefore computed using the reduced eigenvector matrix $\mathbf{A}$:

$$\mathbf{y} = \mathbf{A}'\mathbf{x}$$

B. Factor Analysis

Factor analysis is a statistical technique whose objective is to represent a set of variables in terms of a smaller number of hypothetical variables called factors [22]. It achieves this by analyzing the interrelationship (correlations) among the original data variables and determines whether the observed correlations can be explained by the factors.

The observed variables $\mathbf{x}$ are therefore represented in terms of the factor variables $\mathbf{z}$ that model correlations between variables of $\mathbf{x}$ and matrix $\mathbf{u}$ which accounts for independent noise in each feature of $\mathbf{x}$. The model is given by:

$$\mathbf{x} = \mathbf{\Lambda}\mathbf{z} + \mathbf{u} \tag{11}$$

where $\mathbf{\Lambda}$ is the factor loading matrix whose elements represent how much a factor explains a variable. Factor analysis works on the assumption that the factor variables are normally distributed with zero mean and a unit variance, $\mathcal{N}(0, \mathbf{I})$ and that the random variable $\mathbf{u}$ has the distribution $\mathcal{N}(0, \mathbf{\Psi})$ where $\mathbf{\Psi}$ is a diagonal matrix. According to the model in equation (11), $\mathbf{x}$ is therefore distributed with zero mean and covariance $\mathbf{C}_x = \mathbf{\Lambda}\mathbf{\Lambda}' + \mathbf{\Psi}$. The goal of factor analysis is to find the $\mathbf{\Lambda}$ and $\mathbf{\Psi}$ that best model the covariance structure of $\mathbf{x}$ [23].

Below is a summary of the procedure for factor analysis. The technique makes use of the expectation-maximization algorithm to determine the required $\mathbf{\Lambda}$ and $\mathbf{\Psi}$.

Step 1: Initialization

Initialize Ψ to an identity matrix of size $m \times m$, and Λ to be uniformly distributed $\mathcal{U}(0,1)$ and be of size $m \times d$ where d is the number of desired output components.

Step 2: Expectation step
Calculate expected values of the factors given $\mathbf{x}$:

$$E[\mathbf{z}|\mathbf{x}] = \beta\mathbf{x} \tag{12}$$

where $\beta = \mathbf{\Lambda}'{\mathbf{C}_\mathbf{x}}^{-1}$. Thereafter, calculate the second moment of the factors which measures the uncertainty in the factors, according to

$$E[\mathbf{z}\mathbf{z}'|\mathbf{x}] = \text{var}(\mathbf{z}|\mathbf{x}) + E[\mathbf{z}|\mathbf{x}]E[\mathbf{z}|\mathbf{x}]' \tag{13}$$
$$= (\mathbf{I} - \beta\mathbf{\Lambda}) + \beta\mathbf{x}\mathbf{x}'\beta' \tag{14}$$

where $\mathbf{I}$ is a $d \times d$ identity matrix.

Step 3: Maximization step
Compute the maximum likelihood estimates of $\mathbf{\Lambda}$ and $\mathbf{\Psi}$ as follows:

$$\mathbf{\Lambda}^{\text{new}} = \mathbf{x}E[\mathbf{z}|\mathbf{x}](E[\mathbf{z}\mathbf{z}'|\mathbf{x}])^{-1} \tag{15}$$

$$\mathbf{\Psi}^{\text{new}} = \frac{1}{n}\text{diag}\{\mathbf{x}\mathbf{x}' - \mathbf{\Lambda}^{\text{new}}E[\mathbf{z}|\mathbf{x}]\mathbf{x}'\} \tag{16}$$

where the *diag* operator sets all non-diagonal entries to zero. Next compute the log-likelihood of the factor analysis model,

$$ll = \frac{1}{2}\Big(\log(\det({\mathbf{C}_\mathbf{x}}^{-1})) - \frac{1}{n}\sum\sum({\mathbf{C}_\mathbf{x}}^{-1}\mathbf{x}) * \mathbf{x}\Big) \tag{17}$$

where $*$ denotes element-wise multiplication, and repeat steps 2 and 3 until convergence has been reached.

Step 4: Apply linear mapping
Let $\mathbf{A} = \mathbf{\Lambda}$, the d-dimensional transformed matrix is calculated through the linear combination of the factor loadings and the original dataset as presented below:

$$\mathbf{y} = \mathbf{A}'\mathbf{x} \tag{18}$$

C. Locality Preserving Projections

Locality Preserving Projections (LPP) is a manifold learning algorithm proposed by He *et al.* [24] which aims to discover the meaningful low dimensional structure. He *et al.* present the LPP algorithm as an alternative to PCA as PCA fails to capture underlying data structures that lie on a nonlinear manifold [25]. LPP has the added benefit that it is a linear dimension reduction algorithm unlike other manifold learning algorithms such as ISOMAP [26] which are nonlinear and are computationally expensive.

The projections of the algorithm are obtained by firstly building a graph that incorporates neighbourhood information of the dataset, and then computing a transformation matrix which maps the data points to a subspace using the notion of the Laplacian of the graph.

Step 1: Construct a neighbourhood graph
Let G denote a graph with m nodes which correspond to the number of variables in the original dataset. Using the k-nearest neighbours algorithm, an edge is placed between node i and j, if i is among k nearest neighbours of j and vice-versa. Once the graph is obtained, LPP will attempt to preserve it when choosing projections.

Step 2: Choose weights using a Gaussian kernel
Given graph G, an $m \times m$ weight matrix $\mathbf{W}$ is constructed by assigning a weight W_{ij} based on equation (19) if a connection (edge) exists between node i and j. A weight of zero is assigned if there is no connection between the nodes. This results in the weight matrix being sparse and symmetric.

$$W_{ij} = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{t}} \tag{19}$$

Step 3: Compute eigenvalues and eigenvectors
Compute the Laplacian matrix $\mathbf{L}$ as shown in equation (20),

$$\mathbf{L} = \mathbf{D} - \mathbf{W} \tag{20}$$

where $\mathbf{D}$ is a diagonal matrix whose entries are column sums of weight matrix $\mathbf{W}$, $D_{ii} = \sum_j W_{ji}$.

Thereafter, compute the eigenvalues and eigenvectors for the generalized eigenvector problem:

$$\mathbf{X}\mathbf{L}\mathbf{X}'\mathbf{a} = \lambda\mathbf{X}\mathbf{D}\mathbf{X}'\mathbf{a} \tag{21}$$

The eigenvector decomposition algorithm yields a full matrix $\mathbf{a}$ where the columns correspond to eigenvectors, and a diagonal matrix of generalized eigenvalues, λ. The The column vectors of $\mathbf{a}$ are ordered according to their eigenvalues in ascending order.

Step 4: Apply linear mapping
The transformation vector $\mathbf{A} = (\mathbf{a}_0, \mathbf{a}_1, \cdots, \mathbf{a}_{d-1})$ is then embedded in the linear equation,

$$\mathbf{y} = \mathbf{A}'\mathbf{x}, \tag{22}$$

to output the transformed data matrix $\mathbf{y}$.

IV. Experiments and Results

This section describes the database used and the experimental setup for the comparative analysis of the three linear techniques described in section III. It also also reports the parameter search procedure for the different NARX recurrent neural networks used. The results are presented and discussed in section IV-C.

A. Emotion Database

The emotion database used in this work forms part of the SEMAINE corpus [27] which was recorded to study natural social signals that occur between humans and artificially intelligent agents. It contains audiovisual recordings of humans who interact with four emotionally stereotyped characters - role-played by humans - portraying the following personalities:

(i) even-tempered and sensible, (ii) happy and outgoing, (iii) angry and confrontational, and (iv) sad and depressive. The recordings were filmed at a frame rate of 50 frames per second, with each human-agent interaction session lasting an average of 5 minutes. Due to the high frame rate, a block averaging technique was applied to reduce the samples to 25 frames per second.

The interactions were annonated by two to eight raters in continuous time using continuous values along the dimensions valence, arousal, power and expectation. Only the valence and arousal dimensions are considered in this work.

The dataset was partitioned into a training set of 41 videos and a test set of 18 videos.

B. Experimental Setup

An experiment was designed to determine the optimal dimension size of each feature reduction technique. Each technique was setup to reduce the features to the following dimensions: $10, 20, 40, 60, 80$ and 100. A NARX recurrent neural network was optimized for each feature reduction technique per dimension size. A grid-search was conducted to estimate the NARX-RNN model parameters using five-fold cross-validation with Pearson's correlation coefficient as the evaluation metric. The parameters that had to be optimized were the number of hidden nodes ($n_h \in \{20, 40, 60, 80, 100, 120\}$), the regularization parameter ($\lambda \in \{2^2, 2^4, 2^6, 2^8, 2^{10}\}$), the output time lag ($D_y \in \{1, \cdots, 15\}$). The resulting ouput time lags were 10 and 5 for the arousal and valence dimensions respectively. In addition, a NARX network was trained on all 1593 video features to serve as a baseline model.

The LPP algorithm uses the k-nearest neighbours algorithm to construct its neighbourhood graph. A default value of $k = 12$ was used for all LPP implementations. All the feature reduction techniques were implemented using the Matlab Toolbox for Dimensionality Reduction[1] [28].

C. Results

The feature reduction techniques were evaluated using the average Pearson's correlation coefficient which is obtained by computing the correlation between the emotion predictions and ground truth for each video in the dataset, and then averaging over all videos in a specific emotion dimension. The performance of the investigated techniques was estimated over 30 independent runs. The following subsections present the averages of the 30 independent runs.

1) Optimal dimension sizes: Figure 5 is a plot of the performance of each feature reduction technique over the investigated dimension sizes. The PCA technique outperforms the LPP and FA techniques across all component (dimension) sizes for both the arousal and valence emotions. The correlation results, apart from the FA results, follow a bell-shaped curve indicating that the chosen dimension sizes were sufficient to locate the optimal dimension configurations for each technique.

[1]http://lvdmaaten.github.io/drtoolbox/

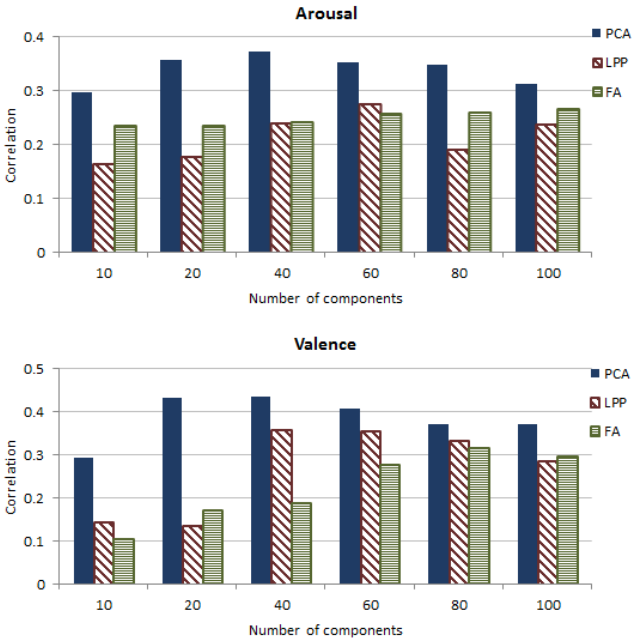

Fig. 5. Correlation results of all the techniques trained with different dimension sizes

Exploring the component make-up of the PCA technique, Figure 6 reveals that the first 20 dimensions explain approximately 50% of the variance while the first 100 dimensions explain 70% of the variance. This reveals that the majority of the 1593-long dimension dataset is not required, and that the chosen dimensions are within Kaiser's eigenvalue rule of thumb which states that the eigenvalues of the components retained should be greater or equal to 1 [29].

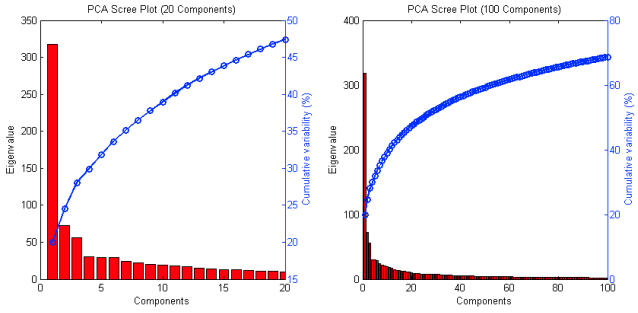

Fig. 6. Component-variance analysis of the PCA technique

2) Technique Comparison: The best performing models for each technique are listed in Table I with the corresponding network configurations and dimension sizes. A statical analysis was conducted using the Mann-Whitney U test at 95% confidence level to determine if the techniques differed significantly. The results of the Mann-Whitney U test indicate that the PCA technique significantly outperformed all other techniques for the arousal and valence emotions, however,

there was no significant difference between LPP, FA and the NARX network with no feature reduction for the arousal emotion. For the valence emotion, it was found that LPP performed better than FA and the no-reduction model, and that FA outperformed the no-reduction model.

TABLE I
COMPARISON OF FEATURE REDUCTION TECHNIQUES

TECHNIQUE	DIM	NARX-n_h	NARX-λ	CORR
AROUSAL				
PCA	**40**	**40**	**4**	**0.372**
LPP	60	40	16	0.275
FA	100	20	4	0.268
No reduction	1593	20	256	0.273
VALENCE				
PCA	**40**	**40**	**64**	**0.436**
LPP	40	20	4	0.359
FA	80	100	4	0.316
No reduction	1593	80	1024	0.220

To remove any performance advantage that may have been caused by having trained the networks with different parameters, another analysis was conducted where 20 components were extracted from each technique and were trained using the same NARX-RNN model. The parameters used for the NARX-RNN models were $n_h = 20$ and $\lambda = 16$ for arousal, and $n_h = 100$ and $\lambda = 64$ for valence. The results of this analysis are presented in Table II. According to the Mann-Whitney U test, the PCA still significantly outperforms other techniques. LPP outperformed FA for the valence emotion, a result that is consistent with the results in Table I. However, LPP was significantly outperformed by FA for the arousal emotion.

These results seem to be consistent with Baek *et al.*'s [30] findings that PCA outperforms factor analysis. Furthermore, van der Maaten *et al.* [28] submitted that from the results they obtained from their comparative analysis of PCA with twelve nonlinear feature reduction techniques, they found that nonlinear techniques are not yet capable of outperforming traditional PCA. However, Shermina's results [25] showed superior performances of the LPP over the PCA. Zhang *et al.* [31] noted that very often feature reduction techniques are problem specific. It is therefore expected to have conflicting reports in different domains.

TABLE II
COMPARISON OF FEATURE REDUCTION TECHNIQUES USING 20 DIMENSIONS AND THE SAME NARX-RNN MODEL

TECHNIQUE	AROUSAL	VALENCE
PCA	**0.357**	**0.432**
LPP	0.181	0.135
FA	0.235	-0.001

D. Data Analysis

The annotation of continuous dimensional emotion is a very challenging task as it is highly subjective, and requires a higher amount of attention and cognitive processing compared to non real-time, discrete annotation tasks [32]. This affects the learning ability of a model as it relies on accurate and consistent ground truth. Therefore an analysis of the data is required to determine whether efficient modeling of emotions is inhibited by unreliable emotion labels or by the shortcomings of the model.

Table III lists the nine groups of raters which consist of two to eight raters. Each group annotated at least one video, with the highest number of videos (26) being annotated by group 7. An inter-rater reliability test was conducted for the database used using intraclass correlation. Intraclass correlation is a measure that assesses the degree of agreement and consistency of raters. The overall results for valence report a consistency of 0.66 and absolute agreement level of 0.55 which indicate low agreement between raters. The consistency and absolute agreement measures for arousal are 0.63 and 0.3 respectively. This challenge is highlighted in [32] and suggestions for better annotation of continuous emotions are tabled there.

TABLE III
INTER-RATER ANALYSIS FOR VALENCE EMOTION LABELS

RATER GROUP	NUMBER OF RATERS	VIDEOS RATED	MEAN CONSISTENCY	MEAN AGREEMENT
Group 1	2	8	0.536	0.498
Group 2	2	3	0.413	0.354
Group 3	3	2	0.744	0.733
Group 4	3	2	0.456	0.393
Group 5	4	1	0.540	0.405
Group 6	5	1	0.863	0.597
Group 7	6	26	0.780	0.670
Group 8	6	4	0.806	0.675
Group 9	8	12	0.786	0.646
AVERAGE RESULTS			0.658	0.552

V. CONCLUSION

This paper comparatively discussed three feature reduction techniques, namely principal component analysis (PCA), locality preserving projections (LPP) and factor analysis (FA) on the problem of continuous dimensional emotion recognition. Various dimension sizes were explored for each technique and a NARX-recurrent neural network was optimized for each technique variant. Experimental results showed that PCA significantly outperformed LPP and FA for both arousal and valence emotion dimensions. The large reduction of features and corresponding better performance confirm that feature reduction is a crucial step for building compact and accurate models, especially for incorporation in human-robot technologies.

Recently, the input modalities of emotion recognition systems have been extended to allow for detection of facial and vocal expressions, gestures and body postures. The multiple modalities often increase the accuracy and robustness of emotion systems since some modalities may carry complementary information. Thus, multiple feature sets representing each modality have to be obtained which leads to a very

large feature space of different forms. Therefore, future work includes exploring dimensionality reduction methods that can transform the multiple feature sets into a unified space of lower dimension. The fusion of multiple kernel learning algorithms with dimension reduction techniques show great promise, and provide a good starting point.

ACKNOWLEDGMENT

This research was supported by the Bradlow Foundation and the South African National Research Fund. The authors would like to thank Nyalleng Moorosi for the helpful discussions that greatly assisted the research and Stefan Gruner for his support.

REFERENCES

[1] C. Nass and B. Reeves, "The media equation: How people treat computers, televisions, and new media as real people and places," 1996.

[2] S. Flach, D. Margulies, and J. Söffner, *Habitus in Habitat I: Emotion and Motion*, ser. Habitus in Habitat. Peter Lang, 2010. [Online]. Available: https://books.google.co.za/books?id=fmEeLoGH-F0C

[3] P. Ekman and W. V. Friesen, "Nonverbal behavior in psychotherapy research." in *Research in Psychotherapy Conference, 3rd, May-Jun, 1966, Chicago, IL, US*. American Psychological Association, 1968.

[4] A. Mehrabian, *Nonverbal communication*. Transaction Publishers, 1977.

[5] D. Grandjean, D. Sander, and K. R. Scherer, "Conscious emotional experience emerges as a function of multilevel, appraisal-driven response synchronization," *Consciousness and cognition*, vol. 17, no. 2, pp. 484–495, 2008.

[6] P. Ekman, *Emotion in the human face*. Cambridge University Press, 1982.

[7] S. Baron-Cohen, *Mind reading: the interactive guide to emotions*. Jessica Kingsley Publishers, 2003.

[8] H. Gunes, "Automatic, dimensional and continuous emotion recognition," 2010.

[9] M. Mortillaro, B. Meuleman, and K. R. Scherer, "Advocating a componential appraisal model to guide emotion recognition," *International Journal of Synthetic Emotions (IJSE)*, vol. 3, no. 1, pp. 18–32, 2012.

[10] P. C. Ellsworth and K. R. Scherer, "Appraisal processes in emotion," *Handbook of affective sciences*, vol. 572, p. V595, 2003.

[11] C. Breazeal and R. Brooks, "Robot emotion: A functional perspective," *Who needs emotions*, pp. 271–310, 2005.

[12] B. Schuller, M. Valster, F. Eyben, R. Cowie, and M. Pantic, "Avec 2012: the continuous audio/visual emotion challenge," in *Proceedings of the 14th ACM international conference on Multimodal interaction*. ACM, 2012, pp. 449–456.

[13] J. D. Moore, L. Tian, and C. Lai, "Word-level emotion recognition using high-level features," in *Computational Linguistics and Intelligent Text Processing*. Springer, 2014, pp. 17–31.

[14] L. Van Der Maaten, "Audio-visual emotion challenge 2012: a simple approach," in *Proceedings of the 14th ACM international conference on Multimodal interaction*. ACM, 2012, pp. 473–476.

[15] T. Baltrušaitis, P. Robinson, and L.-P. Morency, "3d constrained local model for rigid and non-rigid facial tracking," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2610–2617.

[16] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[17] G. Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 6, pp. 915–928, 2007.

[18] E. Diaconescu, "The use of narx neural networks to predict chaotic time series," *WSEAS Transactions on Computer Research*, vol. 3, no. 3, pp. 182–191, 2008.

[19] J. M. P. Menezes and G. A. Barreto, "Long-term time series prediction with the narx network: an empirical evaluation," *Neurocomputing*, vol. 71, no. 16, pp. 3335–3343, 2008.

[20] B. G. Horne and C. L. Giles, "An experimental comparison of recurrent neural networks," *Advances in neural information processing systems*, pp. 697–704, 1995.

[21] N. Banda, A. Engelbrecht, and P. Robinson, "Continuous emotion recognition using a particle swarm optimized narx neural network," in *Affective Computing and Intelligent Interaction (ACII), 2015 IEEE Conference on*. IEEE, 2015, p. to appear.

[22] J.-O. Kim and C. W. Mueller, *Introduction to factor analysis: What it is and how to do it*. Sage, 1978, no. 13.

[23] Z. Ghahramani, G. E. Hinton *et al.*, "The em algorithm for mixtures of factor analyzers," Technical Report CRG-TR-96-1, University of Toronto, Tech. Rep., 1996.

[24] X. He and P. Niyogi, "Locality preserving projections," in *Advances in Neural Information Processing Systems*, 2004, pp. 153–160.

[25] J. Shermina, "Application of locality preserving projections in face recognition," *International Journal of Advanced Computer Science and Applications*, vol. 1, no. 3, 2010.

[26] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[27] G. McKeown, M. F. Valstar, R. Cowie, and M. Pantic, "The semaine corpus of emotionally coloured character interactions," in *Multimedia and Expo (ICME), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1079–1084.

[28] L. J. van der Maaten, E. O. Postma, and H. J. van den Herik, "Dimensionality reduction: A comparative review," *Journal of Machine Learning Research*, vol. 10, no. 1-41, pp. 66–71, 2009.

[29] H. F. Kaiser, "The application of electronic computers to factor analysis." *Educational and psychological measurement*, 1960.

[30] K. Baek, B. Draper *et al.*, "Factor analysis for background suppression," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, vol. 2. IEEE, 2002, pp. 643–646.

[31] Y. J. A. Zhang, L. Qian, and J. Huang, "Monotonic optimization in communication and networking systems," *Found Trends Networking*, vol. 7, no. 1, pp. 1–75, 2013.

[32] A. Metallinou and S. Narayanan, "Annotation and processing of continuous emotional attributes: Challenges and opportunities," in *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*. IEEE, 2013, pp. 1–8.