Facial Expression Recognition with Recurrent Neural Networks

Alex Graves, Jürgen Schmidhuber
Robotics and Embedded Systems Lab, Department of Computer Science
[graves,juergen.schmidhuber]@in.tum.de

Christoph Mayer, Matthias Wimmer, Bernd Radig
Image Understanding and Knowledge-Based Systems, Department of Computer Science
[mayerc,wimmerm,radig]@informatik.tu-muenchen.de

Technische Universität München, Germany

***Abstract*— This paper presents a complete system for automatic facial expression recognition. The Candide-3 face model is used in conjunction with a learned objective function for face model fitting. The resulting sequence of model parameters is then presented to a recurrent neural network for classification. The advantage of using a recurrent network is that the temporal dependencies present in the image sequences can be taken into account during the classification. Since the entire process is automatic, and the recurrent networks can be used to make online predictions, the system would be ideal for real-time recognition. This would make it suitable for the CoTeSys 'coffee break' scenario, where guests must be recognised and served by robot waiters. Promising experimental results are presented on the Cohn-Kanade database.**

I. Introduction

Existing methods for human-machine interaction are often considered unintuitive. As a consequence a lot of time is required for humans to adapt to the operation of a specific machine. In contrast, we aim at granting machines the ability to adapt to typical human behaviour. To participate in natural human-machine interaction, machines must be able to derive information from human communication channels, such as spoken language, gestures or facial expressions.

Model-based image interpretation techniques extract information about human faces. Models impose knowledge about the object of interest and reduce high dimensional image data to a small number of expressive model parameters. This representation of the image content facilitates subsequent recognition and classification tasks. The model used by our system is the deformable 3D wire frame Candide-3 face model [1]. The Candide parameter vector $p = (r_x, r_y, r_z, s, t_x, t_y, \sigma, \alpha)^T$ describes the affine transformation $(r_x, r_y, r_z, s, t_x, t_y)$ and the deformation (σ, α). The 79 deformation parameters indicate the shape of facial components such as the mouth, the eyes, or the eyebrows via a set of 116 anatomical landmarks (see Figure 1). These parameters are estimated with help of learned objective functions as described in our earlier work [2]. This approach ensures accurate and robust estimation of model parameters.

Recurrent neural networks use these features to classify the facial expression. Recurrent networks are able to robustly derive information from sequences of data vectors by exploiting the fact that data represented by successive vectors is also connected semantically and therefore interdependent. We present the classifier data that is extracted from successive images in image sequences displaying facial expressions. Unlike many competing systems, the approach is fully automated and requires no human intervention.

A. Related Work

The task of recognising facial expressions is usually subdivided into three subordinate challenges: face detection, feature extraction, and expression classification [3]. The first step aims at determining the position and shape of the face in the image. Features descriptive for facial expressions or head gestures are extracted in the second step. In the third step a classifier is applied to the features to identify the expression class.

1) Face Model Fitting: Cootes et al. [4] introduced modelling shapes with Active Contours. Further enhancements included the idea of expanding shape models with texture information [5]. Recent research also considers modelling faces in 3D space [1], [6].

Van Ginneken et al. learned local objective functions from annotated training images [7]. In this work, image features

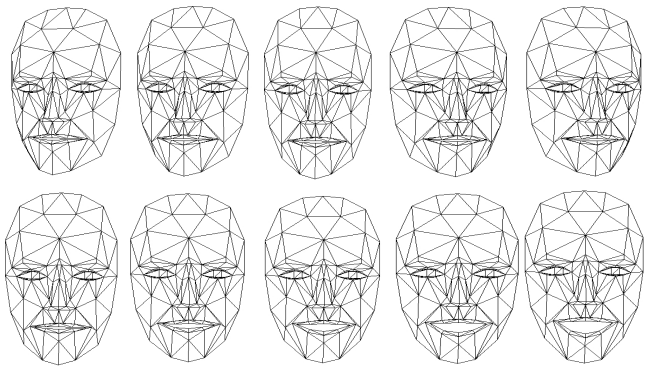

Fig. 1. The Candide-3 face model is able to reflect various face shapes and facial expressions.

are obtained by approximating the pixel values in a region around a pixel of interest The learning algorithm used to map images features to objective values is a k-Nearest-Neighbor classifier (kNN) learned from the data. Romdhani combines the use of a multitude of features with a multi-stage fitting approach to fit 3D face models to 2D images [8]. Each fitting stage leads to more accurate model fits, and different features are used in each stage.

Our methodology combines the two approaches above, in that it uses a multitude of qualitatively different features [8], determines the most relevant features using machine learning [7], and learns objective functions from annotated images [7]. Furthermore, we extend these two approaches, by using machine learning to completely automate the process of determining calculation rules by composing features. More importantly, we formalise properties that *ideal* objective functions have, and enforce these during the generation of training data. This influences the shape of the learned objective function, which is approximately ideal.

2) Feature Extraction: Michel et al. [9] extract the location of 22 feature points within the face and determine their motion between an image that shows the neutral state of the face and an image that represents a facial expression. The very similar approach of Cohn et al. [10] uses hierarchical optical flow in order to determine the motion of 30 feature points. In contrast to our approach, these features are extracted from a sequence of images rather than from single images. Since we use a classifier that inherently considers temporal aspects we do not take this into consideration in the feature extraction step.

3) Feature Classification: The facial expression is determined from the extracted features. Mostly, a classifier is learned from a comprehensive training set of annotated examples. Some approaches infer the facial expression from rules stated by Ekman and Friesen [11]. Kotsia et al. take this approach [12]. Michel et al. [9] train a Support Vector Machine (SVM) that determines the visible facial expression within the video sequences of the Cohn-Kanade Facial Expression Database by comparing the first frame with the neutral expression to the last frame with the peak expression.

B. Organisation of the Paper

The remainder of this paper is structured as follows: in section II we introduce our model fitting approach. In section III we present the features we extract from images and how this differs to related work. In section IV we give details about our classification approach and present its advantages. Section V provides an experimental evaluation of our approach. Finally, Section VI summurizes our approach and introduces future work.

II. Face Model Fitting

In order to extract high-level information, model parameters have to be estimated that best describe the face within a given image. Model fitting solves this task and is often addressed by minimising an objective function that evaluates how well a model parameterisation fits a given image. The objective function is often designed manually, such as the pixel error between the model's rendered surface and the underlying image content. In contrast, we propose to learn the objective function from annotated example images, see Figure 2 [2].

The *objective function* $f(I,p)$ yields a comparable value that determines how accurately a parameterised model p fits to an image I. The *fitting algorithm* searches for the model parameters p that optimise the objective function. However, this paper shall not elaborate on them but we refer to [13] for a recent overview and categorisation.

The objective function, which we consider the core component of the model fitting process, is often designed manually using the designer's domain knowledge and intuition. Afterwards, its appropriateness is subjectively determined by inspecting its result on example images and example model parameters. If the result is not satisfactory the function is tuned or redesigned from scratch [8], [14]. Since the design-inspect loop is iteratively executed, manually designing the objective function is highly time-consuming — see Figure 2 left.

A. Ideal Objective Functions

In contrast, we propose to learn the objective function rather than designing it manually, see Figure 2 right. This approach is based on the general properties of *ideal* objective functions. The key idea behind the approach is that if the function used to generate training data is ideal, the function learned from the data will also be approximately ideal. Furthermore, we provide a large number of image features. The learning algorithm is able to consider all the features together and the resulting objective function allows model fitting with both good runtime performance and high accuracy.

An ideal objective function should have two properties. First, its global minimum should correspond to the correct position of the contour point. Second, it should have no other local minima. Equation 1 depicts an *ideal* objective function $f_n^\star$. It simply computes the Euclidean distance between the correct location $\hat{x}_n^\star$ of the n^{th} contour point and a location u on the image plane. Note that the vector of correct contour points $\hat{x}^\star$ must be specified manually.

The function $f_n^\star$ already shows ideal characteristics. Unfortunately, this function is not feasible for previously unseen images, because it requires the correct locations of the contour points $\hat{x}^\star$, which have to be manually specified. However, our approach uses $f_n^\star$ to generate training data for learning an additional objective function f_n^ℓ that does not require knowledge of $\hat{x}^\star$.

$$f_n^\star(I,u) = |u - \hat{x}_n^\star| \tag{1}$$

B. Learning Objective Functions

We annotate a set of images with the correct contour points $\hat{x}^\star$. For each $\hat{x}_n^\star$, the ideal objective function returns the

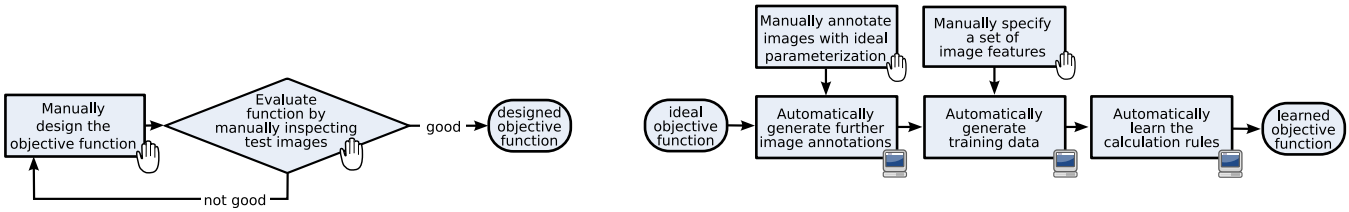

Fig. 2. left: the design approach. right: the learn approach.

minimum $f_n^\star(I,\hat{x}_n^\star) = 0$ by definition. Further coordinate-to-value correspondences are automatically acquired by varying $\hat{x}_n^\star$ along the perpendicular and recording the value returned by the ideal objective function in the second step.

Finally, the calculation rules of the objective function are learned with tree-based regression [16]. The advantage of this machine learning approach is that it selects only the most relevant features from the vast amount of features provided. Therefore, the values of far fewer image features need to be computed during the process of fitting the model which reduces the time requirement.

This approach does not require expert knowledge and is domain-independent. Furthermore it yields more robust and accurate objective functions, which greatly facilitate the task of the associated fitting algorithms. Accurately estimated model parameters in turn are required to infer correct high-level information, such as facial expression or gaze direction. We refer to [2] for a detailed description and evaluation of our approach.

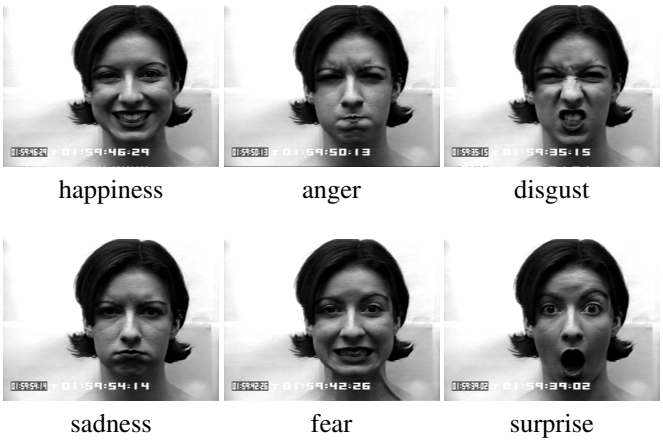

Fig. 3. The six universal facial expressions as they occur in [15].

Fig. 4. The Candide-3 face model consists of 116 landmarks and reflects the face shape by 79 shape parameters.

III. Feature Extraction

Ekman and Friesen [17] identified six universal facial expressions that are expressed and interpreted in the same way by people all over the world. They do not depend on the cultural background or the country of origin. Figure 3 shows one example of each facial expression.

The Facial Action Coding System (FACS) [18] expands on this work to precisely describe the muscle activities within a human face. Action Units (AUs) denote the motion of particular facial parts and state the facial muscles involved. Since our model has been designed with the work of Ekman and Friesen in mind, we infer the facial expression currently visible from the movement of its landmarks.

The deformation parameters (σ,α) describe the constitution of the visible face and the position of the 116 landmarks. The examples in Figure 4 illustrates the relationship between the facial expression and the value of (σ,α). We therefore consider (σ,α) to provide high-level information to the interpretation process. To provide training data for the classification step, we fit the model to every image of the sequence and extract the model parameters. Note, that the extracted features refer to a single image only and could also be obtained for a set of non-dependent images. However, the classifier will exploit the fact that the features are extracted from semantically linked images.

IV. Classification

Given the sequence of extracted features, we then want to classify the facial expression. This can be done directly, by outputting a single class for the entire sequence (as in e.g. [9]), or indirectly, by first classifying the individual frames of the video sequence, then combining these to get the overall expression. Although direct sequence classification generally gives better results, it requires that the video be presegmented into individual expressions. It also makes real-time recognition impossible, since the classification is only available at the end of the sequence. Our system is based on frame classification, and has been designed with unsegmented, real-time applications in mind.

As noted above, facial expressions are inherently temporal. However most classifiers, including support vector machines, decision trees and feedforward neural networks, are designed for static patterns. This means that in order to use them as expression classifiers, the sequence of input features must be preprocessed into a single, fixed-length vector. As well as requiring significant effort on the part of the experimenters,

this approach often throws away significant temporal dependencies.

Recurrent neural networks are a connectionist architecture where one or more of the network layers is connected to itself (illustrated in Figure 5)

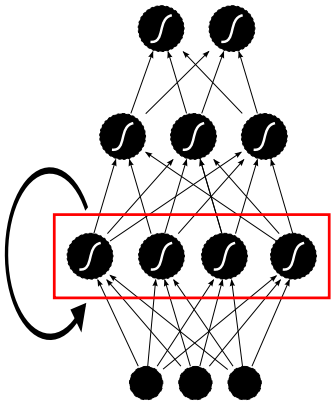

Fig. 5. Recurrent neural network.

The self connections allow the network to build an internal representation of past events, thereby allowing it to make flexible use of temporal context. In addition, the internal representation tends to be more robust to shifts and distortions in the input sequence (e.g. the same expression enacted at different speeds) than static methods.

Since the network is designed for temporal patterns, the sequence of extracted Candide features can be fed to it directly.

One refinement to the basic recurrent architecture is the use of *long short-term memory* (LSTM) [19] cells. As shown in figure 6, LSTM cells use linear units protected by multiplicative gates to store information over long periods of time. This extends the range of temporal context available to the net.

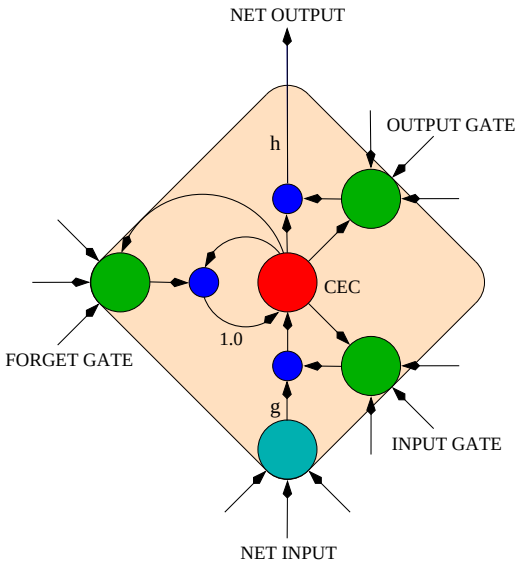

Fig. 6. The long short-term memory architecture.

Another improvement is the use of *bidirectional recurrent networks* [20] to provide future as well as past context. As shown in Figure 7 bidirectional networks scan through the same sequence forwards and backwards with two separate hidden layers, both of which are connected to the same output layer. The output classifications can therefore be made in the light of all surrounding context in the input sequence. However one disadvantage to bidirectional networks is that the entire sequence must be complete before they can be used, which makes them unsuitable for real-time applications.

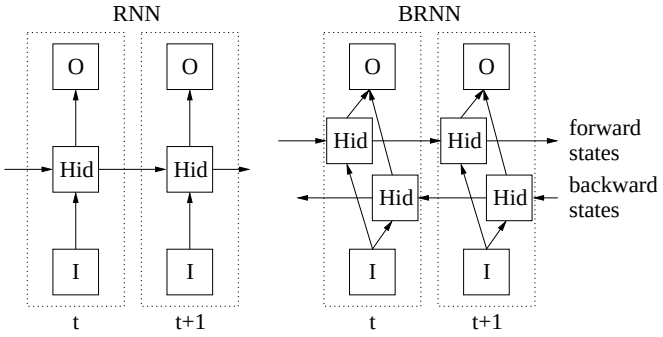

Fig. 7. Normal recurrent network (RNN) and Bidirectional recurrent neural network (BRNN).

In the following experiments we evaluate both bidirectional LSTM [21] and unidirectional LSTM for facial expression classification.

V. Experimental Evaluation

We evaluated our system on the Cohn-Kanade facial expression recognition database [15]. The task was to classify each of the video sequences into one of the six standard expression classes: happiness, anger, disgust, sadness, fear and surprise. Following the methodology used in [9], we used 5-fold cross validation to determine the classification accuracy.

A. Data

The Cohn-Kanade Facial Expression Database is publicly available and referenced by several research groups [22], [12]. It contains short image sequences of about 100 persons of both genders with age varying between 18 and 30 years. The image sequences start with the neutral face and evolve into the peak expression. All images are taken from a frontal view. The database also includes specifications for the activation of AUs and the facial expression depicted.

B. Classifier Parameters

We used two networks, a bidirectional LSTM network with 100 LSTM cells in both the forward and backward hidden layers, and a unidirectional LSTM network with 150 cells in the hidden layer. Both networks had 35 input units and six output units, one for each target class. All hidden layers were completely connected to themselves, and to the input layer and output layer. The bidirectional network had

Classification error rate on the Cohn-Kanade database.

Classifier	Mean Error Rate
bidirectional LSTM	14.6 $\pm$ 0.9%
unidirectional LSTM	18.2 $\pm$ 0.6%

110,606 trainable parameters in total, while the unidirectional network had 112,956.

The output layer had a softmax activation function and was trained with the cross-entropy objective function for classification. During training, each frame of each video was classified separately according the overall expression in the video (there were no classification targets for the first third of each video, since these were determined to be expressionless). Once training was complete, the accuracy of the network was assessed by summing up over the frame classifications, then normalising to achieve a probabilistic classification for the entire sequence

The inputs were normalised to have mean 0 and standard deviation 1. Gaussian noise with standard deviation 0.5 was added to them during training to improve generalisation.

To calculate the 5-fold classification score, the data was divided into 5 test sets, each containing 20% of the exemplars of each of the six expression classes. For each test set, 15% of the remaining sequences were used as validation sets for early stopping, and the rest were used for training. The network weights were randomly according to a Gaussian distribution with a mean of 0 and a standard deviation of 0.1. Each of the 5 experiments was repeated 10 times, to account for the effects of random initialisation. The error rate quoted in the results is the mean over all 50 experiments $\pm$ the standard error.

C. Results

Table V-C shows the classification results on the Cohn-Kanade database. As can be seen, the bidirectional network gave significantly better performance. However the unidirectional net is interesting in that it could be used for real-time recognition.

Note that in both cases the training set was easily learned by the classifier, with virtually all sequences correctly classified. This indicates that the key difficulty of the task lies in generalising from the training to the test examples. In particular, generalisation is difficult with such a small number of training samples. We would expect a substantial gain in performance if a larger dataset was used.

VI. Conclusions and Future Work

We present an approach for facial expression estimation that combines state-of-the-art techniques for model-based image interpretation and sequence labelling. Learned objective functions ensure robust model-fitting to extract accurate model parameters. The classification algorithm is explicitly designed to consider sequences of data and therefore considers the temporal dynamics of facial expressions. Future work aims at presenting the classifier training data that is obtained from various publicly available databases to reflect a broader variety of facial expressions. Furthermore, our approach will be tuned towards applicability in real-time. It is planed to create a working demonstrator from this approach.

References

[1] J. Ahlberg, "Candide-3 – an updated parameterized face," Linköping University, Sweden, Tech. Rep. LiTH-ISY-R-2326, 2001.

[2] M. Wimmer, F. Stulp, S. Pietzsch, and B. Radig, "Learning local objective functions for robust face model fitting," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 30, no. 8, 2008.

[3] M. Pantic and L. J. M. Rothkrantz, "Automatic analysis of facial expressions: The state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1424–1445, 2000. [Online]. Available: citeseer.ist.psu.edu/pantic00automatic.html

[4] T. F. Cootes and C. J. Taylor, "Active shape models – smart snakes," in *Proceedings of the 3rd British Machine Vision Conference*. Springer Verlag, 1992, pp. 266 – 275.

[5] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," in *5th European Conference on Computer Vision*, H. Burkhardt and B. Neumann, Eds., vol. 2. Freiburg, Germany: Springer-Verlag, 1998, pp. 484–498.

[6] V. Blanz, K. Scherbaum, and H. Seidel, "Fitting a morphable model to 3d scans of faces," in *Proceedings of International Conference on Computer Vision*, 2007.

[7] B. Ginneken, A. Frangi, J. Staal, B. Haar, and R. Viergever, "Active shape model segmentation with optimal features," *IEEE Transactions on Medical Imaging*, vol. 21, no. 8, pp. 924–933, 2002. [Online]. Available: citeseer.ist.psu.edu/vanginneken02active.html

[8] S. Romdhani, "Face image analysis using a multiple feature fitting strategy," Ph.D. dissertation, University of Basel, Computer Science Department, Basel, CH, January 2005.

[9] P. Michel and R. E. Kaliouby, "Real time facial expression recognition in video using support vector machines," in *Fifth International Conference on Multimodal Interfaces*, Vancouver, 2003, pp. 258–264.

[10] J. Cohn, A. Zlochower, J.-J. J. Lien, and T. Kanade, "Feature-point tracking by optical flow discriminates subtle differences in facial expression," in *Proceedings of the 3rd IEEE International Conference on Automatic Face and Gesture Recognition*, April 1998, pp. 396 – 401.

[11] P. Ekman and W. Friesen, *The Facial Action Coding System: A Technique for The Measurement of Facial Movement*. San Francisco: Consulting Psychologists Press, 1978.

[12] I. Kotsia and I. Pitaa, "Facial expression recognition in image sequences using geometric deformation features and support vector machines," *IEEE Transaction On Image Processing*, vol. 16, no. 1, 2007.

[13] R. Hanek, "Fitting parametric curve models to images using local self-adapting seperation criteria," PhD thesis, Dep of Informatics, Technische Universität München, 2004.

[14] D. Cristinacce and T. F. Cootes, "Facial feature detection and tracking with automatic template selection," in *7th IEEE International Conference on Automatic Face and Gesture Recognition*, Southampton, UK, April 2006, pp. 429–434.

[15] T. Kanade, J. F. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis." in *International Conference on Automatic Face and Gesture Recognition*, France, March 2000, pp. 46–53.

[16] R. Quinlan, "Learning with continuous classes," in *Proceedings of the 5th Australian Joint Conference on Artificial Intelligence*, A. Adams and L. Sterling, Eds., 1992, pp. 343–348. [Online]. Available: http://citeseer.ifi.unizh.ch/quinlan92learning.html

[17] P. Ekman, "Universals and cultural differences in facial expressions of emotion," in *Nebraska Symposium on Motivation 1971*, J. Cole, Ed., vol. 19. Lincoln, NE: University of Nebraska Press, 1972, pp. 207–283.

[18] ——, "Facial expressions," in *Handbook of Cognition and Emotion*, T. Dalgleish and M. Power, Eds. New York: John Wiley & Sons Ltd, 1999.

[19] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[20] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, pp. 2673–2681, November 1997.

[21] A. Graves and J. Schmidhuber, "Framewise Phoneme Classification with Bidirectional LSTM and Other Neural Network Architectures," *Neural Networks*, vol. 18, no. 5-6, pp. 602–610, June/July 2005.

[22] R. Schweiger, P. Bayerl, and H. Neumann, "Neural architecture for temporal emotion classification," in *Affective Dialogue Systems 2004, LNAI 3068*. Kloster Irsee: Elisabeth Andre et al (Hrsg.), June 2004, pp. 49–52.