A Survey paper on Facial Expression Synthesis using Artificial Neural Network

Deepti Chandra
Shri Shankaracharya Technical Campus, Bhilai
Chhattisgarh

Rajendra Hegadi
Kruti Institute of Technology & Engineering, Raipur
Chhattisgarh

Sanjeev Karmakar
Bhilai Institute of Technology (BIT) Durg
Chhattisgarh

ABSTRACT
Facial expressions are a kind of nonverbal communication. They carry the state of emotion of a person. Facial expression plays an important role in face-to face human-computer communication. Automatic facial expression synthesis became popular research area nowadays. It can be used in many areas such that physiology, education, murder squad, analysis of tendency to crime to get a clue about mental signals of a person. Although considerable efforts have been made to enable computers to speak like human beings, how to express the rich semantic information through facial expression still remains a challenging problem. This paper presents a novel approach using artificial neural network. This paper proposes two different approaches with different methods for facial expressions synthesis based on artificial neural networks (ANN). Firstly, Modeling using Hidden Markov Models is proposed. Secondly, modeling using Recurrent Neural.

Keywords
Emotional Facial Expression Modeling, Face Synthesis, Facial Animation, Hidden Markov Models, Recurrent Neural Networks.

1. INTRODUCTION
With the development of CAD/CAM software, modeling of mechanical objects becomes easier than ever before. On the other hand, non-mechanical objects remain a problem both in shape modeling and motion control. Usually, the shape of a non-mechanical object is irregular and asymmetric . A typical shape modeling method for such objects is free from deformation, which requires tedious work from an expert artist or modeler. When non-affine transformation is applied, the control parameters should be reset for each model. As one of the representative object, we discuss animated human face modeling in this paper .Many research efforts have been focused on the achievement of realistic representation of human face since the pioneer work of Parke [15]. However, the irregular shape of the head, the complex facial anatomical structure and various facial tissue behaviors make it still a formidable challenge in computer graphics. The challenge of modeling an animated human face can be described from several parts as follows:

Shape modeling: To create a human head model for a specific person is a tedious task because of the variety of facial features and appearance. The development of the 3D range scanner helps to capture the shape information of complex object. But holes or gaps may appear due to the variant reflective properties on facial surface, overlapped or folded surfaces produced by merge procedure may result in visual artifacts. For human head, hairy surface cannot be appropriately recognized by laser scanner. Lips are not separated and eyes are not recognized as an individual part of human head. These noisy and incomplete data increases the processing difficulty.

Expression synthesis: Differing from body animation, the human facial expression cannot be synthesized by skeleton-skinning techniques. Point-based approach does not help a lot either because it does not provide enough controls for the skin dynamics. The facial expression is driven by muscle contractions. The muscle forces are applied on the interior layer of soft tissue. The soft tissue has a multi-layer structure, which consists of properties such as visco elasticity, incompressibility and nonlinearity. These properties result in dynamic stress-strain behavior. The various thickness of skin tissue at different regions also increases the complexity of the dynamics. Because of the dynamic behavior, bulge and wrinkles can be noticed among expressions. Facial motion from one expression to another cannot be simply described as linear interpolation between two expressions, either.

Photo-realistic rendering: To mimic human face in the real world under specific light condition, advanced rendering technique and shading model should be applied on the face model. These techniques include texture mapping, normal mapping, shadow casting and ray tracing, etc. The challenges here include high-resolution full head texture creation, an appropriate shading model to represent the resolution and refraction on the skin and high-efficiency model to represent skin bumps.

Real-time performance: Based on the aforementioned challenges, it can be understood that to create an accurate dynamic personalized human expression synthesis. sizer with photo-realistic skin appearance, comprehensive computational model will be employed. However, getting an interactive updating rate is a practical requirement for applications such as virtual surgery training, video conference and human-computer interactive virtual avatar. This leaves us challenge to balance between the interactive performance and synthetic quality. Taking advantage of the modern hardware such as Graphic Processing Unit (GPU), Ray-tracing Processing Unit (RPU) and Physics Processing Unit (PPU) and merging into a practical system is also a research topic.

1.1 Applications
There is sufficient proof of the great vital force of computer facial animation and its extensive application. In the next decade, computer facial animation will be indispensable.

Entertainment: There are increasing interest of putting virtual characters in films and videos (Shrek Series, Final Fantasy VII: Advent Children 2005, etc.,). Even in movies acted by real human beings, computer-aided facial morphing

is a highly demanded technique. Computer games always try to mimic a "real" world in purpose of player experience. In movie industry, high quality realistic human motion is normally the essential concern. Oppositely, in game industry real-time performance is the key factor to consider.

Virtual Surgery: Surgeons expect to get the realistic human brain model so they can make an anticipation to reduce the risk of surgery to a relatively lower point. An intern can get trained on the virtual surgery system before they do the real operation. These kinds of systems always require an accurately rendered facial model and the anatomically correct tissue reaction. Volume rendering technique is normally used to display different structure in human head. Koch [6] designed a virtual surgery model using Finite Element Model (FEM). Maciel et al. [8] also modeled the biological behavior of soft tissue using FEM to assist orthopedic diagnosis and surgery planning.

Video Conference: Traditionally, in video conference system, we need to transfer the whole video data of the face. This requires a stable high-rate broadband. In a computer-synthesized animation system, the animation control parameters will be transferred so that a narrow-band video conference system can be expected. Eisert [30] has developed an advanced video conference system using MPEG-4 which encodes high quality head and shoulder motion sequences at a bit rate about 1-kbit/s. The low bandwidth requirements of the system make it easier to integrate wireless devices such as PDAs, cellular phones and notebooks into a video-based communication system. Similarly, Fedorov et al. presented their system in [13].

Lip Reading: For the physically challenged, such as the deaf people, lip reading is important to communicate with normal people. Even for normal people, lip reading will enhance the comprehension of sentence in a noisy environment. For xample,

ViSiCAST [33] is a project that translates English text to the motion of virtual avatar and provides services for deaf citizens.

Education: Text-to-speech talking head is also suitable for kids and adults to learn how to speak a special word in a sentence. Furthermore, using motion capture mechanism a comparison between the trainee and the source can be achieved and comment will be made. Virtual Human Interface [21] is a product designed by Digital Elite Inc. which delivers information to the end user by photo-realistic animated characters. It supports *What, Where, How* forms of communication paradigm and can be used for interactive education, communication and marketing.

1.2. Emotions and Emotional Facial Expressions

Emotions are definitively part of our life and condition our behaviors, feelings, reactions to events and to people. Emotions have been widely studied from different perspectives by many researchers in several fields (anthropology, sociology, psychology, cognitive science, philosophy, computer science). Although the definitions of emotion diverge considerably, most researchers agree that emotions are a process with various components, such as physiological responses (visceral and muscular states), autonomic nervous system and brain responses, memories, feelings verbal responses, and facial expressions. From a theoretical point of view emotions can be distinguished in four classes: a) Primary Emotions; b) Secondary Emotions; c) Tertiary Emotions; and d) Basic Emotions.

a. **Primary Emotions** are innate, produced by reactive mechanisms mapping external stimulus patterns to reorganized behaviors, enabling fast reactions to environmental changes. Primary emotions are those that we feel first, as a first response to a situation. They are instinctive responses. For example, the reaction (hiding or ducking the head) to objects flying overhead at a certain speed is a typical primary emotion ([19], [14].

b. **Secondary Emotions** (e.g. anxiety, sorrow) are learned associations between recognized stimulus patterns generated by primary emotions and analyzed situations where these patterns occurred [29]. The processes involved in learning and analysis of these situations are named "deliberative mechanisms". These mechanisms are cognitive processes taking into consideration goals, belief, standards and expectations, enabling reasoning about situations, plan making, and understanding of action consequences [12].

c. Tertiary Emotions (e.g. shame, guilt, envy) might be consequences of meta-management mechanisms that enable the cognitive awareness of internal processes or states and provide the possibility to reason about these internal states and processes. Tertiary emotions are "cognitive perturbances" arising from goal conflicts in an information processing architecture [15].

d Basic Emotions (e.g. happiness, fear, surprise) are discussed by Ortony and Turner [14]. They mean "basic" as psychologically and biologically primitive. Psychologically basic emotions aim to build a set of "psychologically irreducible emotions". That means that these emotions would be building blocks for other emotions. Biologically, basic emotions are innate and serve survival functions kept through evolution. One consequence of this argument is that basic emotions should be found across human cultures and across species of higher animals. The main arguments in favor of the existence of basic emotions are presented by Ekman [11] and Izard [8], who show that facial expressions of basic emotions are universally recognized. Emotions are linked to facial expressions in some undetermined loose manner ([21], [19]).

Emotions are linked to facial expressions in some undetermined loose manner ([20], [22]). Emotional facial expressions are the facial changes in response to a person's internal emotional states, intentions, or social communications.

2. MODELING APPROACHES

In the last years, many works of face modeling have been carried on, more for analysis rather than synthesis, and in the last case more for speech synthesis than facial synthesis. Among these, only someone models emotional facial expressions. In most of the early face synthesis systems, such facial expressions were modeled by rules (symbolic approach). Among these, some of the most relevant works are the following:

2.1.1 Emotional facial expressions

Humans express a lot of emotions and do so in several ways (by means of speech, gaze, facial muscles, head motions, gestures, etc.). In order to reduce the modeling space we decided not to consider all possible emotions but only a limited set and in particular conditions. We oriented our attention to approaches based on "basic emotions". Among several possible solutions in this area, we choose the Ekman's

set ([3] an[33). This set includes six basic emotions (called also "the big six") consisting of anger, disgust, fear, happiness, sadness, and surprise. Such emotions are considered "basic" for two main reasons: a) according to a Darwinian perspective, they represent survival-related patterns of responses to events in the world which have been selected over the course of our evolutionary history; b) all other emotions are thought to be somehow derived from them by combinations and mixtures [34]. This set of emotions has also a "universal" valence established by evolutionary, developmental, and cross-lingual studies .

Also other researchers advocate the concept of "basic emotions"(some alternatives to the Ekman's set are summerised in [11) and propose similar or bigger sets of emotions, according to different theories

2.1.2 Modeling dynamics

The dynamic aspect of emotional expressions is related to their temporal course. As stated by Ekman in [4],[5], and described in [12] and [10], each emotional expression is characterized by three temporal phases :

onset: it is the time interval in which the expression, starting from a neutral state, reaches maximal intensity;

apex: it is the time interval during which the expression maintains its maximal intensity;

offset: it is time interval in which the expression, starting from maximal intensity, returns to neutral state.

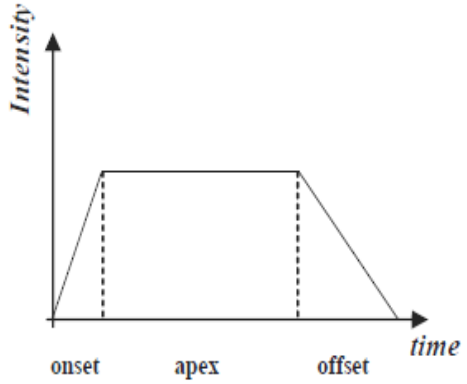

Fig 1: Temporal course of an emotional expression

In other words, a single expression can have different expressivity depending on the manner it appears (onset), the time it remains on the face (apex) and finally the speed it disappears (offset). For the sake of simplicity, let assume that is possible to show different emotional states having exactly the same intensity4 but different temporal evolution. This case is depicted in Figure 2 where there are different trapezoidal functions, with same height but different onset, apex and offset values.

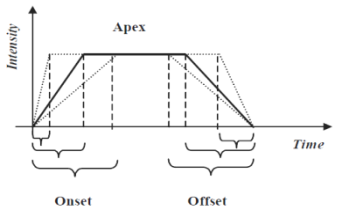

Fig.2: Emotional expressions with different Onset, Apex and Offset

By comparing temporal courses of different emotional facial expressions it becomes evident that different emotions have different speeds of generation. So, for example, "fear" and "surprise" have an onset shorter than other emotions, while "sadness" is typically slower and its offset is longer [5].

2.2. Connectionist and Markovian Models

The dynamics of the facial expressions through time can be formally described as a discrete-time sequence of random feature vectors, drawn from a distribution in a properly defined feature space. In other words, we are facing a random-process modelling problem in which some kinds of statistical inference, or learning, has to be accomplished from a corpus of training data samples.

The most popular machine learning approaches to sequence modeling rely on Hidden Markov Models (HMMs), as well as on Artificial Neural Networks (ANNs). The former are rooted in statistics, on the basis of a maximum-likelihood parametric estimation assumption. ANNs, on the contrary, are a "universal" non-parametric estimator, trained from labeled samples via application of the gradient method. While HMMs are effective in several applications, e.g. speech recognition, their generative capabilities cannot be fully exploited within the present scenario,as we will see shortly. In this respect, a particular ANN topology,referred to as the Recurrent Neural Net (RNN), will help to a significant extent. The following Sections shortly describe HMMs and ANNs from a theoretical viewpoint, introducing their architectures, principles and fundamental algorithms. For a detailed discussion see [8], [20].

2.2.1 Hidden Markov Models

An HMM is a pair of stochastic processes: an hidden Markov chain and an observable process which is a probabilistic function of the states of the former. This means that observable events in the real world (e.g. *xyz* marker coordinates) are modeled with (possibly continuous) probability distributions, that are the observable part of the model, associated with individual states of a discrete-time, first-order Markovian process. The semantics of the model (conceptual correspondence with physical phenomena)

is usually encapsulated in the hidden part: for instance, in Acoustic Speech Recognition (ASR) an HMM can be used to model a word in the task-dependent vocabulary, where each state of the hidden part represents a phoneme (or sub-phonetical unit), whereas the observable part accounts for the statistical characteristics of the corresponding acoustic events in a given feature space (e.g. a sampled acoustic signal, represented in a proper way).

More precisely, an HMM is defined by:

1. A set *S* of *Q* states, **S** = {S1, …., SQ}, which are the distinct values that the discrete, hidden stochastic process can take.
2. An *initial state* probability distribution, i.e π={ {Pr (Si|t π= 0), Si ∈ S}, where *i* is a discrete time index.
3. A probability distribution that characterizes the allowed transitions between states that is is **a**ij = {Pr (Sj at time t|Si at time t-1), Si ∈ S} where the transition probabilities are assumed to be independent of time *t.*
4. observation or feature space F, which is a discrete or continuous universe of all possible observable events

(usually a subset of of of $\Re_d$,, where d is the dimensionality of the observations).

5. A set of probability distributions (referred to as emission or output probabilities) that describes the statistical properties of the observations for each state of the model:

$$bx = \{bi(x) = Pr(x \_Si), Si \in S, x \in F\}$$

HMMs represent a learning paradigm, in the sense that examples of the event that is to be modeled can be collected and used in conjunction with a training algorithm in order to *learn* proper estimates of π, a and b_x.

These algorithms belong to the class of *unsupervised learning* techniques, since they perform unsupervised parameter estimation of the probability distributions without requiring any prior labeling of individual observations (within the sequences used for training) as belonging to specific states. Once training has been accomplished, the HMM can be used for *decoding* or *recognition*. Training and decoding algorithms suffer from some major intrinsic limitations (see [15]). In short, the classical HMMs rely on strong assumptions on the statistical properties of the phenomenon at hand. For instance, the stochastic processes involved are modeled by first-order Markov chains, and the parametric form of the probability density functions that represent the emission probabilities associated with all states is heavily constraining. In addition, the number of parameters in HMMs does strongly limit their implementability, and increases the model complexity (in time and space). Given these limitations, the use of ANNs with their generative output capabilities, their capability to perform non-parametric universal estimation over whole sequences of patterns, and their limited number of parameters appeared definitely promising.

2.2.2 Artificial Neural Networks

For the sake of simplicity, let us start introducing a feed-forward connectionist model trainable with supervision by providing, for each training input sample, the corresponding desired output (more sophisticated models will be described in the next Sections). This means that input vectors are presented at the input of the network one at a time. Based on the input, the network computes an output vector.

Let us consider the training training $T \{(x , y) \mid k\ 1,...N\}\ k\ k$ == where $x_{1},..., x_N$ are N input samples with their corresponding desired (target) outputs $N\ y ,..., y\ 1$The aim is to build a model able to compute for each input of the training data an output close to the desired one, according to some optimality criterion. A particular family of such models is represented by Simple Linear Perceptrons (SLPs).

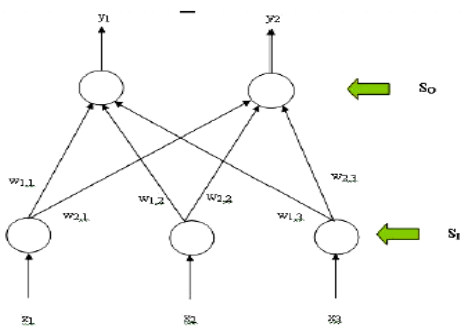

Fig 3: Simple Linear Perceptron

An input signal is propagated forward along the connections, and multiplied by the corresponding connection weight. All the incoming weighted signals to a given output unit are summed together, to form the input to the unit itself. The unit reacts by producing an activation response which is equal to its input. This model is usually referred to as a layered network, with the obvious meaning that units are arranged into subsequent layers: the computation proceeds from one layer to the next in bottom-upborder, but never in a lateral or backward manner. For this reason the network is called "feedforward".The family of models considered in this section fits the training data as summarized by the following equation (written for the generic i-th output component):

$$\hat{y}_1(x) = \sum_{R^- S_I} w_{t.f} x_I$$

This is a homogeneous linear transformation, but an additive bias can be easily added to the model. SPL can thus be seen as linear regression models or linear discriminators for classification. Nevertheless, the way these networks learn from the training set is quite general, and can be extended to the study of other, more complex, ANN architectures.

Once a training set T and a SLP are given, with the obvious assumption that the number of input and output units matches the dimensionality of the input and target vectors, respectively, the learning problem can be stated as the search for the network weights w which optimally fit the data, according to a certain criterion. The latter is usually expressed as a functional of the training data (and of the model) that represents a gain to be maximized or a loss (or risk) to be minimized. A common choice for the criterion function is the sum of squared differences between target outputs and actual outputs, as expressed:

$$C = \frac{1}{2} \sum_{n=1}^{N} \sum_{k \in S_o} (y_{kn} - \hat{y}_{kn})^2$$

Where ykn is the k-th component of the n-th target, $kn\ y\hat{}$ is output of the k-th output unit when the network is presented with the nth input vector, and the multiplicative factor 1/2 is introduced for computational convenience, as we shall see below. The minimization of 3-2 is known as the least square criterion. A general and broadly used optimization technique for the minimization of expression 3-2 is the gradient descent method. Most network training algorithms are based on it. Although it is not guaranteed that the approach will eventually reach the global minimum of the criterion function, these techniques often produce practically useful behavior. Gradient descent iterative algorithms are used also for training more complicated ANN architectures

2.2.3 Multi-layer Perceptrons

The most popular neural network architecture is the MultiLayer Perceptron (MLP), also known as feed-forward neural network.This is an extension of the SLP with additional layers of units,called hidden layers

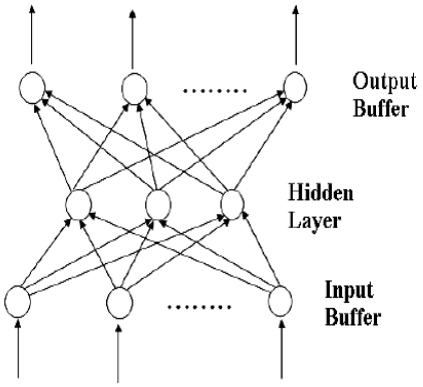

Fig 4: Multi-layer Perceptron

In the hidden layers an internal, intermediate representation of data is formed. An MLP is fed by an input vector and subsequent computations are passed from layer to layer in the usual feedforward manner. This produces an output vector of values of the units of the last (output) layer of the network. Activation functions associated to hidden and output units can be linear or non-linear, and can be different for different units. Input units still act as placeholders for the components of the current input vector. A training algorithm estimates a set of weights to be assigned to the connections between each pair of units belonging to adjacent layers, in order to optimize a training criterion.

Training is generally supervised. Let us consider a training set $T = \{(x_k, y_k) | k = 1, ..., N\}$ The learning problem for MLPs is to find the weights that result in a (generally non-linear) model that best fits the training data, given a certain criterion function

The most common choice for the criterion is the sum of squared differences between target and actual outputs, just as in the case of SLPs. An on-line gradient descent technique [15] is used to minimize

The cost function:

$$C = \frac{1}{2} \sum_{k \in S_o} (y_{kn} - \hat{y}_{kn})^2$$

computed after the presentation of a certain input pattern x associated with a desired output vector y of a layered network with a set of input units $I S$, hidden units $H S$, and output units $O S$. Extension to more hidden layers is straightforward. The learning algorithm is similar to the one for SLPs. weight change $_{ij}\ \Delta w$ of the connection strength between the j-th hidden unit and the i-th output unit is computed as follows:

$$\Delta w_{if} = -\eta \frac{\partial C}{\partial w_{if}}$$

2.2.4 Neural Nets for Sequence Processing: TDNNs and RNNs

There are problems, such modeling facial movements, in which samples are temporal sequences of patterns, instead of individual independent samples. To take these temporal dependencies into account, two major classes of neural networks have been proposed, namely Time-Delay Neural Network, and Recurrent Neural Networks. While the former is still an MLP with units fed by an input value and a number of its predecessors, the latter generalizes the basic feed-forward architecture of MLPs by allowing arbitrary connections between units, e.g. loops and backward connections.

Time-Delay Neural Networks

Time-Delay Neural Networks (TDNNs), also known as tapped delay lines, represent an effective attempt to train a static MLP for time-sequence processing, by converting the temporal sequence into a spatial sequence over corresponding units. As shown in Figure , the input layer has been enlarged to accept as many input patterns as the (fixed) sequence length to be processed at each time step. Input vectors enter the network from the leftmost set of input units. At each time step, inputs are shifted to the right through the unit delay line that links each set of input units to the right-adjacent one, and the next input pattern is fed into the leftmost position. The same extension can also be applied to subsequent layers, introducing tapped-delay mechanism between hidden units (e.g. only the first block of units in the tapped line actually receives input from the previous layer), giving the ability to deal with more complicated time dependencies.

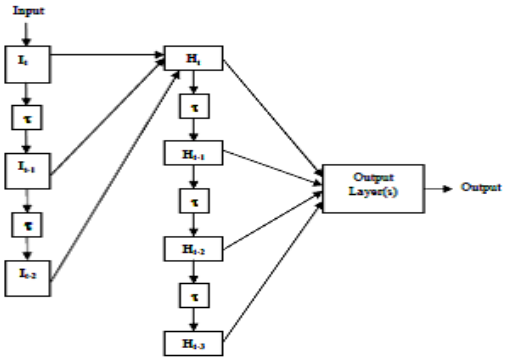

Fig 5: The BP algorithm can be used to train such a network.

2.2.5 Recurrent Neural Networks

Recurrent Neural Networks (RNN) provide a powerful extension of feed-forward connectionist models by allowing connections between arbitrary pairs of units, independent of their position within the topology of the network. Self-recurrent loops of a unit onto itself, as well as backward connections to previous layers, or lateral links between units belonging to the same layer are all allowed

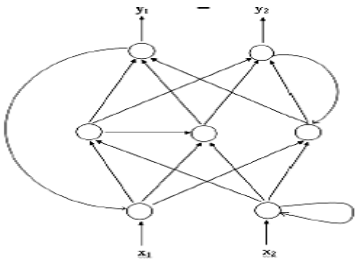

Fig 6: Recurrent Neural Networks (RNN)

RNNs behave like dynamic systems. Once fed with an input, the recurrent connections are responsible for an evolution in time of the internal state of the network. RNNs are particularly suited for sequence processing, due to their ability to keep an internal trace, or memory, of the past. This memory is combined with the current input to provide a context-dependent output. A first type of RNNs is applied to static patterns (i.e., the input is fixed) and its dynamic converges to specific attractors. For instance, in Boltzmann Machines [34]

the recurrent connections are symmetric (bidirectional propagation of signal is allowed along the connection, i.e. pairs of adjacent units have an influence on each other). Limitations of Boltzmann Machines reside in the requirement of symmetric (non-directional) recurrent connections, and in the considerable computation time required to perform the simulated annealing. In addition, although they are historically and conceptually relevant, they are practically not feasible for sequence processing.

Another family of RNNs is sometimes referred to as partially recurrent nets. They were introduced in [33] and resulted in a wide range of applications in sequence processing (both in recognition and in generation).

In this case the basic architecture is that of a standard MLP, with the addition of a set of recurrent connections from the units in a given layer to the corresponding units of a previous layer (or in the same layer). Recurrent connections propagate the signal back to the units of one of the layers of the MLP, or to an additional context or state layer. The units that receive signal from the recurrent connections act either as pre-processors, filtering the current (forward propagated) input with the previous signal, or as a register that keeps a memory of previous history. The weights of the recurrent connections are generally fixed and set equal to a constant, chosen in order to calibrate the amount of previous information to be taken into account. The standard BP algorithm is used to train the underlying MLP architecture, but without the computation of the full gradient on the parameters, since the effect of the past activities through recurrent connections is not taken into account

A training method for general recurrent architectures is now briefly described. It can be easily derived from the standard BP algorithm for feed-forward networks. In spite of its apparent simplicity, this technique is quite effective whenever the length of the sequences to be learned is not too large, and we are willing to wait for the end of a sequence before updating parameters. This is often the case when a whole training set with many sequences is available, i.e. when no on-line learning is required. The algorithm is called Back-Propagation through Time (BPTT) or unfolding in time [12]

3. MODELING USING HIDDEN MARKOV MODELS

HMMs are one of the most widespread statistical tools to model sequences of data. One of their distinctive features lies in the fact that they can handle sequences of varying sizes, through the use of an internal state variable. They may be used in a *generative framework* in which a different HMM for each class of data to be modelled is defined. In the present framework, HMMs have been used as follows. At first we defined an HMM for each emotional state and viseme. An ideal, simplified HMM topology is depicted in Figure 3.

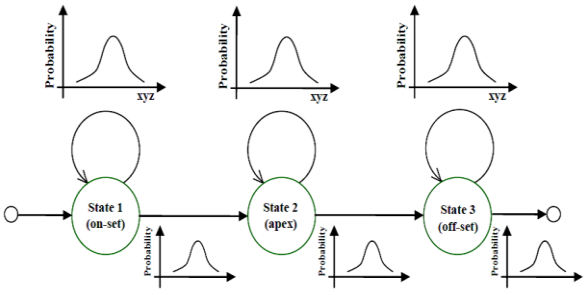

Fig 7: An example of HMM for an emotional state.

This HMM features continuous-density emission probability distributions, associated with the state transitions of the model. It has only 3 states, corresponding, for example, to On-set, Apex,Off-set . The input observations are represented by the vectors of *XYZ* marker coordinates . models were required in order to obtain suitable performance in the present scenario. So, different families of HMMs have been defined for each individualemotional state that has to be modeled. Then, within each such a family, different left-to-right HMMs were introduced for modeling individual visemes6. This modeling includes longer leftto- right hidden Markov chains and herein Mixture of Gaussian *pdfs* have been used to model the emission probabilities. Training samples, clustered by emotional states (without any distinction of intensity) were used, along with the Baum-Welch algorithm, to estimate the model parameters. Due to the continuous nature of the feature space (namely, the space of *XYZ* vectors), a CDHMM was required.

According to the original plan, the intensity modelling would have been introduced in a second step, enlarging the set of HMMs, namely defining an HMM for each emotion-intensity

(E.g. Happiness-Low, Happiness-Medium, Happiness-High,Surprise-Low, Surprise-Medium, Surprise-High, etc.). As well as,on the other side, the modeling of the transition from an emotional state to another one would have been done by the concatenation of corresponding HMMs of the two basic individual emotional expressions

However, unfortunately there are severe drawbacks in the application of HMMs as a facial expression generator: (i) the HMM makes a local stationarity assumption within each one of its states, that is not matched in this scenario; (ii) the underlying Markovian assumption, as well, does not hold for *XYZ* trajectories in feature space: there may be quite long-term time dependencies between *xyz* vectors, i.e. the stochastic process at time *t* does not depend only on the state at time *t-1*, ..., *t-n*; (iii) the reduction of the process to a discrete and finite state chain is arbitrary and not natural: a continuous space of states would better fit the task at hand; (iv) even if the generative mode is made available from the simulator, its application does suffer from computational complexity problems, mostly due to the need of a Monte Carlo-like simulation in order to generate the random quantities associated with the probabilistic parameters of the HMM. The resulting process is so slow, on average, that its real-time application (that is required in the present task, given the final goal of application of this modeling in a talking head) is not feasible in practice.

3.1. Modeling using Recurrent Neural Nets

As alternative to HMMs we considered ANNs. The choice was guided by the fact that it is well-known that they are particularly suited for sequence processing, given their ability to keep an internal trace (memory) of the past.

At the beginning we considered the use of TDNNs because they are effective under several circumstances but, after a thorough study, it was evident that they cannot be used in the present modeling task. The reason is twofold: first of all, they can handle only a limited portion of the time sequences involved in the process (while long-term time dependencies have to be taken into account, as discussed above in the HMM framework). Then, TDNNs can hardly be used in a generative manner: they require a window of input frames, and they yield the corresponding, individual output. This is feasible during the training step, but it is not realistic during the test (i.e.,

during the application of the model on the field), since no such a window of input frames is available at test time

Afterwards, the choice fell on RNNs, for several reasons, in this particular paradigm, the memory of neural nets is combined with the current input to provide a context-dependent output. RNNs behave like dynamical systems. More specifically, RNNs are a viable alternative to HMMs and TDNNs that mostly tackle their corresponding limitations. They can act as generative models: actually, if backward connections are introduced, they enter a self-feeding loop in which discrete-time outputs are a function of an evolving internal state of the RNN, without even needing any further inputs. The length of the generated sequence is not constrained a-priori, i.e. they can be easily used to yield sequences of any required length. No strong statistical assumptions are made. Furthermore, the internal state of RNNs is described in terms of a set of activation values for the recurrent neurons: since activation functions are real-valued, the state itself varies with continuity and it is not constrained within a finite/discrete set. In other words, they are "infinite state" machines. Finally, they are universal models that may take long term time dependencies into consideration (although training the RNN to model such dependencies via gradient-descent is known to be difficult.

4. CONCLUSIONS

The dynamics of the facial expressions through time can be formally described as a discrete-time sequence of random feature vectors, drawn from a probability distribution in a properly defined feature space. The modeling of such dynamics can be accomplished by facing a random-process modeling problem in which some kinds of statistical inference are carried out from a corpus of training data samples. Besides intrinsic limitations of this technique, we found out that, in the present scenario, HMMs suffer from other drawbacks. These concern their specific generative behavior in relation to the facial expression synthesis (where a realistic facial animation is sought). As a matter of fact, the experiments showed that the trajectories (in the *xyz* space) generated by HMMs, albeit satisfying in modeling the prototypical behavior of an emotional facial expression, result in a sort of "piecewise" animation In alternative to HMMs we considered ANNs, and in particular RNNs. The choice was guided by the fact that ANNs are nonparametric, universal approximates. In addition, RNNs are suitable for sequence processing tasks, carrying out estimation over whole sequences of patterns, and they may be used in a generative fashion. From a theoretical point of view, RNNs appeared to be a promising approach to the present synthesis framework, but in practice they turned out to be limited (as the experiments showed). In fact, although RNNs tackled the HMM major drawbacks (in particular, they overcome the HMM piecewise behavior at animation time), yielding smoother and more realistic expressions, after a short period of time these expressions became flat (i.e., their dynamics tend to exhaust and to reach a steady state). This behavior was due to their limited capability in dealing with long-term time sequences.

5. REFERENCES

[1] B. Abboud, F. Davoine, and M Dang. Expressive Face Recognition and Synthesis. In *Proceedings* of *IEEE CVPR workshop on Computer Vision and Pattern Recognition forHuman Computer Interaction*, Madison, U.S.A., 2003.

[2] J. Ahlberg. Extracting MPEG-4 FAPS from Video. In I.S. Pandzic, and R. Forchheimer, editors, *MPEG-4 Facial Animation – the Standard, Implementation and Applications*,John Wiley & Sons, 2002.

[3] E. André, T. Rist, S. van Mulken, M. Klesen, and S. Baldes. The Automated Design of Believable Dialogues for Animated Presentation Teams. In S. Prevost J. Cassell, J. Sullivan and E. Churchill, editors, *Embodied Conversational Characters*. MIT press, Cambridge, MA, 2000.

[4] G. Ball and J. Breese. Emotion and Personality in a Conversational Agent. In S. Prevost J. Cassell, J. Sullivan and E. Churchill, editors, Embodied Conversational Characters. MITpress, Cambridge, MA, 2000.

[5] K. Balci. Xface: Open Source Toolkit for Creating 3D Faces of an Embodied Conversational Agent. In *Proceedings of Smart Graphics*, 2005.

[6] K. Balci. Xface: MPEG-4 based Open Source Toolkit for 3D Facial Animation. In *Proceedings of Advance Visual Interfaces*, Bari, Italy, 2004.

[7] J. Beskow, L. Cerrato, P. Cosi, E. Costantini, M. Nordstrand, F. Pianesi, M. Prete, and G. Svanfeldt. Preliminary Cross-cultural Evaluation of Expressiveness in Synthetic Faces. In E. Andrè, L. Dybkiaer, W. Minker, and P. Heisterkamp, editors, Affective Dialogue Systems ADS'04, Springer Verlag, 2004.

[8] E. Bevacqua, M. Mancini, and C. Pelachaud. Speaking with Emotions. *AISB 2004 Convention: Motion, Emotion and Cognition*. Leeds, United Kingdom, 2004.

[9] T. Bickmore, and J, Cassell. Social Dialogue with Embodied Conversational Agents. In J. van Kuppevelt, L.Dybkjaer, and N. Bernsen, editors, *Advances in Natural,Multimodal Dialogue Systems.*, Kluwer Academic Publishers, 2005.

[10] V. Blanz, C. Basso, T. Poggio, and T. Vetter. Reanimating Faces in Images and Video. In *Proceedings of EuroGraphics'03*, 2003.

[11] M. Brand. Voice Puppetry. In *Proceedings of ACM SIGGRAPH'99*, 1999.

[12] T.D. Bui, D. Heylen, M. Poel, and A. Nijholt. Generation of Facial Expressions from Emotion Using a Fuzzy Rule Based System. In *Proceedings of the 14th Australian Joint Conference on Artificial Intelligence (AI 2001)*, Adelaide, Australia, 2001.

[13] J. Cassell, H. Vilhjalmsson, and T. Bickmore. BEAT the Behavior Expression Animation Toolkit. In *Proceedings of SIGHHRAPH 01*, 2001.

[14] J. Cassell, T. Bickmore, L. Cambell, H. Vilhjalmsson, and H. Yan. Human Conversation as a System Framework: Designing Embodied Conversational Agents. In J. Cassel, J. Sullivan, S. Prevost, and E. Churchill, editors, Embodied Conversational Agents, MIT Press, 2000.

[15] J. Cassell, M. Stone and H. Yan. Coordination and Context dependence in the Generation of Embodied Conversation.In *Proceedings of First International Conference on Natural Language Generation*, 2000.

[16] J. Cassell, J. Sullivan, S. Prevost, and E. Churchill, editors,Embodied Conversational Agents. MIT Press, 2000.

[17] E.S. Chuang, H. Deshpande, and C. Bregler. Facial Expression Space Learning. In *Proceedings of Pacific Graphics'02*, 2002.

[18] I. Cohen, A. Garg, and T. Huang. Emotion Recognition from Facial Expressions using Multilevel HMM. 2000.

[19] M.M. Cohen, and D.W. Massaro. Modelling Coarticulation in Synthetic Visual Speech. In N. Magnenat-Thalmann, and D. Thalmann, editors, *Models and Techniques in Computer Animation*, Springer-Verlag, 1993.

[20] M.M. Cohen, D.W. Massaro, and R. Clark. Training a Talking Head. *Proceedings of 4th IEEE International Conference on Multimodal Interfaces (ICMI'02)*,Pittsburgh, PA, 2002

[21] J. Cohn, K. Schmidt, R. Gross, and P. Ekman. Individual Differences in Facial Expression: Stability over Time, Relation to self-reported Emotion, and Ability to inform Person Identification. In *Proceedings of the International Conference on Multimodal User Interfaces* (ICMI 2002),

[22] T.F. Cootes, and C.J. Taylor. Statistical Models of Appearance for Computer Vision, Wolfson Image Analysis Unit, Imaging Science and Biomedical Engineering, University of Manchester, Manchester M13 9PT, U.K.,2001.

[23] R.R. Cornelious. Theoretical Approaches to Emotion. In *Proceeding of ISCA Workshop on Speech and Emotion*, Belfast, 2000.

[24] E. Cosatto, J. Ostermann, H.P. Graf, and J. Schroeter. Lifelike Talking Faces for Interactive Services. In *Proceedings of the IEEE*, volume 91, number 9, 2003.

[25] P. Cosi, A. Fusaro, D. Grigoletto, and G. Tisato. Data-Driven Tools for Designing Talking Heads Exploiting Emotional Attitudes. In *Proceedings of Tutorial and Research Workshop "Affective Dialogue Systems",* Germany, 2004.

[26] Cosi P., Fusaro A., Tisato G. LUCIA a New Italian Talking-Head Based on a Modified Cohen-Massaro's Labial Coarticulation Model. In Proceedings of Eurospeech 2003, Geneva, Switzerland, 2003.

[27] E. Costantini, F. Pianesi, and P. Cosi. Evaluation of Synthetic Faces: Human Recognition of Emotional Facial Displays. In E. Andrè, L. Dybkiaer, W. Minker, and P. Heisterkamp, editors, Affective Dialogue Systems ADS '04, Springer-Verlag, 2004.

[28] M. D'Amico, and G. Ferrigno. A Technique for the Evaluation of Derivatives from Noisy Biomechanical Data by a Model-Based Bandwidth-Selection Procedure. In Med. & Biol. Eng. & Comp., 28, 1990.

[29] X.-B. Gao, B. Xiao et. al. A Comparative Study of Three Graph Edit Distance Algorithms. Foundations on Computational Intelligence (Edited by A. Abraham, Aboul-Ella Hassanien et al.), ISBN: 978-3-642-01535-9, Springer, Vol. 5, SCI 205, pp. 223-242, 2009.

[30] X.-B. Gao, W. Lu et al. Image Quality Assessment: A Multiscale Geometric Analysis based Framework and Examples. Handbook of Natural Computing (Edited by Rozenberg Grzegorz, Back Thomas H.W., Kok Joost N), ISBN: 978-3-54092911-6, Springer-Verlag, 2011

[31] C. Deng, X.-B. Gao et al. Robust Image Watermarking Based on Feature Regions. Multimedia Analysis, Processing and Communications(Edited by W. Lin, D. Tao, J. Kacprzyk, Z. Li, E. Izquierdo and H. Wang), ISBN: 978-3-642-19550-1, Springer-Verlag Berlin Heidelberg 2011, SCI 346, pp.111-137.

[32] B. Xiao, X.-B. Gao et al. Recognition of Sketches in Photos. Multimedia Analysis, Processing and Communications (Edited by W. Lin, D. Tao, J. Kacprzyk, Z. Li, E. Izquierdo and H. Wang), ISBN: 978-3-642-19550-1, Springer-Verlag Berlin Heidelberg 2011, SCI 346, pp.239-262.

[33] Salvador E. Ayala-Raggi, Leopoldo Altamirano-Robles and Janeth Cruz-Enriquez, "Face Image Synthesis and Interpretation Using 3D Illumination-Based AAM Models", New Approaches to Characterization and Recognition of Faces, InTech, 2011.

[34] Mingli Song , Dacheng Tao ; Shengpeng Sun ; Chun Chen ; Jiajun Bu "Joint Sparse Learning for 3-D Facial Expression Generation" Image Processing, IEEE Transacation,2013

[35] ,Qming Hou,Kun Zhou," Displaced dynamic expression regression for real-time facial tracking and animation", ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2014

[36] Jun li,Weiwei Zu,Zhaiqun CheinLightweight wrinkle synthesis for 3D facial modeling and animation ,Elesvier ,pp117-122,2015,2015