A COMBINED NEURAL NETWORK AND CONTOUR METHOD FOR MOUTH IMAGE LOCATION FOR SPEECH-DRIVEN IMAGE ENHANCEMENT

S-H Luo and R.W.King

Speech Technology Research Group
Department of Electrical Engineering
The University of Sydney

ABSTRACT - This paper describes a new and effective mouth locating method to locate automatically the mouth shape in a human head with shoulder image. This work forms part of our research aimed at to improving the quality of image compression for very low bit rate videotelephony where the motion of the mouth should correspond exactly to what is being uttered, and is not excessively smoothed by any more general purpose data compression method.The paper outlines how we intend to integrate phonetic information with the mouth shape and motion.

INTRODUCTION

It is of current interest in speech and image processing to focus on the developing of videophone and videoconference systems operating over very low bit rate channel (Whybray, 1990). Compared to other video utilities, videophone and videoconference are intended primarily for person-to-person or group-to-group audiovisual communications. It is desirable to operate these at low data rates, ideally at the 64 kbits/s rate offered by a single ISDN channel. Existing video coders operating at this rate can exhibit very poor performance where there is some degree of motion. Accordingly, so-called 'intelligent' image coding (Kaneko, 1991) has recently come in for a great deal of attention for future videophone and videoconference services. Compared to conventional coding techniques, which were designed to transmit the waveform signals, intelligent image coding methods utilize knowledge about the shape and structure of objects and images, and to some extent handle the meaning or content of visual information (Aizawa, 1987, Welsh,1990,Walden et al., 1977).

There exists much mutual information between the acoustic speech signal (speech) and the 'visual' speech signal (mouth movement). In normal speech perception, acoustic speech is reinforced to some degree by observation of the speaker's mouth (Massaro, 1987). Visual speech signals are used in lipreading; a complementary process is to use parameters derived from acoustic speech signals to drive an image of mouth movement. Such a process would provide intelligent enhancement of visual image compression methods and provide the basis for image animation by intelligent coding. Different mouth dimensions and motion (including tongue and teeth position) correspond to different speech utterances. For example, vowels in English differ primarily in terms of the visible tongue height and tongue advancement. Walden (1977) has described an experiment in which all the English consonants were divided into 9 visually discriminable categories, which provide the basis of any technique for speech-enhanced image coding.

To generate mouth movement from speech, as is required in intelligent video coding, a crucial requirement is an automatic and accurate mouth locating method. The method we give here can meet this demand. Mouth location from a head and shoulders image is most conveniently performed in two similar steps: first, the location of the head, and secondly, location of the mouth, using data derived from the head boundary.

Two major processes are undergone in both of these two steps. In the first process, by using an active contour model, called 'snake' (Kass, 1988; Waite & Welsh, 1990), which originates from the way the contour changes its shape), we can derive a shape which has a high probability of being a head or mouth contour. The snake achieves this by means of minimizing three defined 'energy' measures: the image energy, the elastic energy due to snake's stretching, and another elastic energy due to snake's bending. Since the snake computation is a local energy minimizing process, its result may be a head/mouth shape or not. At this stage we introduce a trained multi-layer perceptron net as a pattern recognizer to determine the validity

of the head or mouth location. This first judges if the snake has caught the head or mouth, and then readjusts the snake parameters until the head and mouth contours are found. The paper provides a number of examples of the operation of these processes on experimental image data.

SEEKING A METHOD THAT CAN LOCATE HEAD AND/OR MOUTH BOUNDARY

In a head and shoulder image, it is a very simple task for a human being to locate the head and mouth position and describe it. But this is not a simple task for a computer program. In the image processing and recognition field, the most commonly used methods to distinguish objects from their background exert some edge enhancement or thresholding operations on the given image. Several classes of edge operations exist (Sahoo, 1988); including, for example, gradient operation, Laplacian operator, zero-crossing operation, and morphological edge operation. These operations are effective for images with very explicit edges such as a picture of a rectangle. On a head image, these methods perform rather poorly.

In the course of seeking a head and mouth boundary locating method, the following idea arose: can we initially set a curve in the image, and let the curve be driven by some particular forces which are related to the image so the curve can exactly rest on the contour required? Such an 'active contour' or snake method was introduced by Kass (1988) and has been the subject of attention for head boundary location (Waite & Welsh, 1990). The snake can be used to locate head and mouth boundaries if properly initialised.

Snake is a method of which attempts to provide some of the post-processing that our own visual system performs. It is a continuous curve that attempts to position itself dynamically from a given starting position in such a way that it clings to edges in the image. Snake is a active contour model, changing its shape by means of minimizing some specially defined energy, and in doing so, the curve slithers, giving the method its name.

Mathematically the snake consists of curves that are piecewise polynomials, i.e., the complete curve is in general constructed from N segments,each segment is represented in 2-D plane as $\{x_i(r),y_i(r)\}$, where, $i=1,2,...,N$, $x_i(r)$ and $y_i(r)$ are functions of parameter r.

Referring the snake curve as $u(r)=(x(r),y(r))$, where, r varies between 0 and 1, we can define the snake's energy as,

$$E_{snake} = \int_0^1 E_{snake}(u(r))\,dr = \int_0^1 \{E_{image}(u(r)) + E_{stret}(u(r)) + E_{bend}(u(r))\}\,dr$$

where, E_{image} represents image energy, the result of exerting some gradient operation on the image; E_{stret} represents the elastic energy due to the snake's stretching; E_{bend} is another energy due to the snake's bending.

Given these three properties, a snake builds into itself three important characters: continuity, smoothness and to some extent the capability to fill in sections of an edge that have been occluded. Figure 1 shows an experimental result of the snake successfully locating a head, where the outer circle is the initial snake position, and the curve along head boundary gives the final position of the snake driven by these forces.

USING SNAKES TO DERIVE A CANDIDATE HEAD OR MOUTH CONTOUR

After the snake has been given the three properties, an object is chosen when the energy function is minimized. To do this, we need first to derive the representation of E_{image}, E_{stret} and E_{bend}.

Since E_{image} is the measure of image strength, an edge detecting operation can be operated on the image and the result acts as E_{image}. Here we choose Sobel operation (Abdou & Pratt, 1979).

E_{stret} is proportional to the square of how much the curve is being stretched at that point, so

$E_{stret} = \beta(r)\left(\left(\frac{dx}{dr}\right)^2 + \left(\frac{dy}{dr}\right)^2\right)$, where, $\beta(r) > 0$, represents the amount of elasticity.

E_{bend} is proportional to the curvature of the curve,so

$E_{bend} = \alpha(r)\left(\left(\frac{d^2x}{dr^2}\right)^2 + \left(\frac{d^2y}{dr^2}\right)^2\right)$ where, $\alpha(r) > 0$, represents the amount of stiffness.

So a snake is settled when the minimum of following expression is reached:

$$I(x(r),y(r)) = \int_0^1 \left(\alpha(r)\left(\left(\frac{d^2x}{dr^2}\right)^2 + \left(\frac{d^2y}{dr^2}\right)^2\right) + \beta(r)\left(\left(\frac{dx}{dr}\right)^2 + \left(\frac{dy}{dr}\right)^2\right) - F(x(r),y(r))\right)dr \qquad \text{(EQ 1)}$$

where F(x(r),y(r)) is the result of exerting Sobel operator on the image.

In practice, the minimizing operation is calculated locally. Equation (1) will reach its minimum when the following two independent Euler-Lagrange equations and their corresponding boundary conditions are satisfied:

$$\frac{d^2}{dr^2}\left[\alpha(r)\frac{d^2x}{dr^2}\right] - \frac{d}{dr}\left[\beta(r)\frac{dx}{dr}\right] + \frac{1}{2}\frac{\partial F}{\partial x} = 0 \; ; \quad \frac{d^2}{dr^2}\left[\alpha(r)\frac{d^2y}{dr^2}\right] - \frac{d}{dr}\left[\beta(r)\frac{dy}{dr}\right] + \frac{1}{2}\frac{\partial F}{\partial y} = 0 \qquad \text{(EQ 2)}$$

Equations (2) can be solved by finite difference. Here the solution $(\underline{x}, \underline{y})$ is considered as periodic, i.e.,$(x_0, y_0)=(x_N, y_N),(x_{-1}, y_{-1})=(x_{N-1}, y_{N-1}),(x_{N+1}=x_1, y_{N+1}=y_1),(x_2=x_{N+2}, y_2=y_{N+2})$.

The resulting curve $(\underline{x}, \underline{y})$ can be written as algebraic equations:

$$K\underline{x} = f(\underline{x}, \underline{y}) \; , K\underline{y} = g(\underline{x}, \underline{y}) \qquad \text{(EQ3)}$$

where, $\underline{x} = (x_1, x_2, \ldots, x_N)$; $\underline{y} = (y_1, y_2, \ldots, y_N)$;K: the coefficient matrix;

$$f(\underline{x}, \underline{y}) = (f_1, f_2, \ldots, f_N)^T, f_i = -\frac{1}{2}\frac{\partial F}{\partial x} \qquad ; g(\underline{x}, \underline{y}) = (g_1, g_2, \ldots, g_N)^T, g_i = -\frac{1}{2}\frac{\partial F}{\partial y}$$

Since equation (3) consists a non-linear system, it can be solved using following iteration:

$$K\underline{x}^{(n+1)} - f(\underline{x}^{(n)}, \underline{y}^{(n)}) = t(\underline{x}^{(n+1)} - \underline{x}^{(n)}) \; ; \; K\underline{y}^{(n+1)} - (\underline{x}^{(n)}, \underline{y}^{(n)})g = t(\underline{y}^{(n+1)} - \underline{y}^{(n)}) \qquad \text{(EQ 4)}$$

where, $t > 0$, is a step size; $\underline{x}^{(n)}$ $(\underline{y}^{(n)})$ is the $\underline{x}$ $(\underline{y})$ value after n time iterations.

The iteration operation will proceed until the difference of both $\underline{x}^{(n+1)}$ to $\underline{x}^{(n)}$ and $\underline{y}^{(n+1)}$ to $\underline{y}^{(n)}$ are small enough. So each time when the equation (4) gets its solution, snake will settle down and the local energy minimizing is found. It is evident that if the snake is initially put close enough to head or mouth boundary and the parameters are chosen properly, solution $(\underline{x}, \underline{y})$ will give the contour of head or mouth.

USING ANN AS PATTERN RECOGNIZER TO GUIDE THE SNAKE

As indicated above, snake is an active contour model and can lock on nearby edges. In its practical implementing,there exist several important features that should be attended to and are vital to finding successfully the desired contour. These features include the choices of $t, \alpha(r), \beta(r)$ and initial position.

There is no quantitative method to choose these initial features properly; we can only use some qualitative guidance to direct the parameter choice. Each time the snake arrives at its equilibrium state, human intervention is needed to readjust the parameters to let snake hug on the desired contour.

For a fully automated process, some kind of pattern classifier or recognizer is needed to judge the preliminary shape specified by snake, and automatically adjust its parameters until the real mouth location is found. Pattern classification and recognition are usually achieved in two stages. First, a suitable set of features with desired invariance properties is extracted from the patterns selected for recognition. Here, these properties are invariant under translation, rotation and scale transformation. Then, these features are presented to a

classifier whose purpose is to partition the space of features into decision regions corresponding to each pattern class. Several artificial neural networks (ANN) have been proposed for shape classification (Perantonis & Lisboa, 1992; Lippman, 1987). Unlike traditional classifiers which tend to test competing hypotheses sequentially, the neural network classifiers tests the competing hypotheses in parallel. It has advantages of high computational rate, robustness and the ability to adapt and learn.

Our ANN classifier is a multi-layer perceptron net with the N-points normalized snake samples as its input. The ANN classifier judges all the shape obtained from the snake operation and reinitialize the snake if the previous result is not ideal.

MULTI-LAYER PERCEPTRON NETWORK(MLP) AND ITS IMPLEMENTING TO GUIDE SNAKE

An artificial neural network is specified by three factors: topology of the network, the characteristics of the nodes, and the processing algorithm. The topology of a MLP is a structured hierarchical layered network. A four layer MLP with N inputs, N nodes in the first hidden layer, H nodes in the second hidden layer and M outputs is depicted in Figure 5. The nodes are relatively simple processing elements and the capabilities of MLP stem from the nonlinearities used within them. The nonlinear function most commonly used is the logistic activation function (Rumelhart et al., 1987):

$$f(\alpha) = \frac{1}{1+e^{-\alpha}}$$

The algorithms for multi-layer perceptron network processing can be divided into two phases: retrieving and training. In the retrieving phase, information flows from the input layer through the hidden layers to the output layer. The nodes update their own activation values based on the system dynamics. The network weights and node bias values are determined from a prior training process using a training set of several input and corresponding desired output patterns.

We choose a 4-layer MLP as classifier for head shape recognition and a 3-layer MLP for mouth shape recognition. In using these nets to guide the snakes, the selection of the input normalizing method has a critical role to the performance of the algorithm. In order to enable a classifier to be invariant under translation, rotation and scale transformation, we generally feed the classifier with normalized input. There exist several shape representing methods which are independent of translation, scaling and rotating, for example, Fourier descriptors (Persoon & Fu, 1977), invariant moments (Zakaria, 1987). After detailed study and comparing these methods, we developed a simple and practical normalizing procedure in which the specific utilization for mouth location is considered.

EXPERIMENTAL RESULTS

The strategy we used in our mouth locating method employs the trained MLP net as a recognizer to guide the snake: if the contour caught by snake is not a desired mouth shape, the snake is reinitialized and released until a shape is approved. Although the choice of parameters $t, \alpha(r), \beta(r)$ and snake's initial position are all important to the success of snake, guiding the snake according to all these factors would be very complicated. At this stage, we just consider readjusting snake with different initial positions under the condition that the other parameters are chosen properly.

After the head locating MLP net has been trained with several images, it can easily locate head boundary for both the trained images and new images. The MLP net can rule out the results when snake is trapped on external features near the head, such as a collar.

The mouth locating MLP net uses the result from the head locating net: each time the head locating net works out a head shape, the mouth locating net guides the snake's activity until a mouth shape is found. In deciding whether a shape is a real mouth boundary, some knowledge of mouth's relative position to other face features has been considered. For example, if the width of a shape is greater than 3/4 of the width of head, this shape is disallowed as a mouth contour. Figure 3 gives a example of a real mouth boundary con-

sidered by the training process. Figure 4 gives a mouth boundary recognized by the trained net. In order to test the method's adaptability to translation, scaling and rotation, different staring points are chosen and the results are quite satisfied.

CONCLUSIONS

Locating the mouth in a video image is but the first stage in a process of speech-driven intelligent image coding. More specific location and dimensioning of the of the lips, teeth and tongue are also needed. Then, and this is the main focus of our current attention, the variations in mouth positions for phoneme classes need accurate computation. Currently we are investigating 9 phoneme classes. These classes are classified according to different positions of lips, tongue, teeth, jaw and mouth when they are uttered. In due course, these could be identified by a suitable speech recognition process, and be integrated into the acoustically driven image coding process.

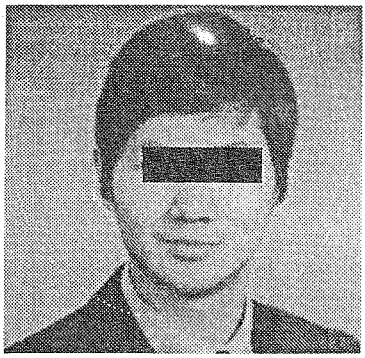

Figure 1. Example of snake successfully locating head

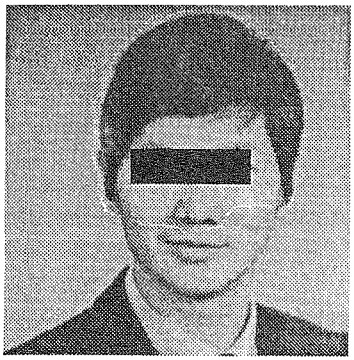

Figure 2. When $\beta(s)$ is chosen larger than necessary, snake penetrates through head boundary

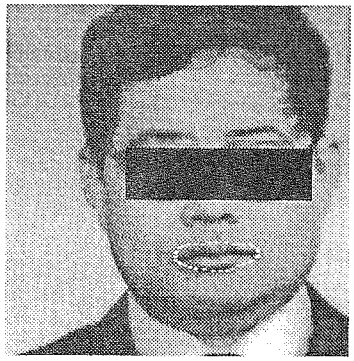

Figure 3. A mouth boundary caught by training process

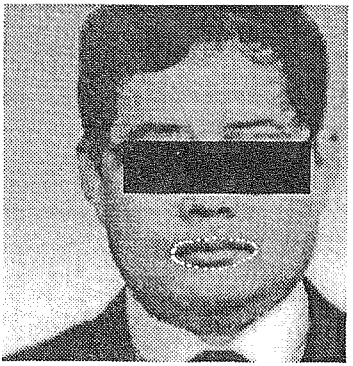

Figure 4. A mouth boundary caught automatically by our method

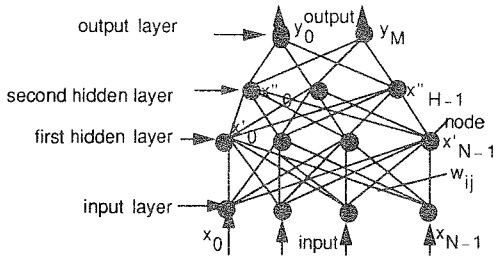

Figure 5. A four layer multi-layer perceptron network

REFERENCES

Abdou, I.E. & Pratt, W.K (1979) *Quantitative design and evaluation of enhancement/thresholding edge detectors*, proc.IEEE 67,pp 753-763,May.

Aizawa, K.,etc. (1987) *Model-based synthesis image coding system-modeling a person's face and synthesis of facial expressions*, GLOBECOM 87,Dec.

Kaneko, M. (1991) *Intelligent image coding-technology for new visual communication services*, Konnichiwa,No.82.

Kass, M., etc.,(1988) *Snakes:active contour models*, International Journal of Computer Vision,pp321-331.

Lippman, R.P. (1987) *An introduction to computing with neural nets*, IEEE ASSP Magazine,April.

Massaro,D.W.(1987),*Speech perception by ear and eye*,Lawrence Erlbaum Assoc.

Perantonis, S.J & Lisboa, P.J.G. (1992) *Translation,rotation,and scale invariant pattern recognition by high-order neural networks and moment classifiers*, IEEE trans on Neural Networks, Vol 3 No 2,March.

Persoon, E. & Fu, K.S. (1977) *Shape discrimination using Fourier descriptors*, IEEE Trans.Syst. Man.Cybern. 7.

Rumelhart, D.E., etc.(1987) *Learning internal representations by error propagation*, Chapter 8 of Vol 1 in Parallel Distributed Processing by Rumelhart and Mcclelland, MIT Press,Cambridge,Mass.,USA.

Sahoo,P.K., etc. (1988) *A survey of thresholding techniques*, Computer Vision, Graphics &Image Processing, Vol 41.

Waite, J.B. & Welsh, W.J. (1990) *Head boundary location using snakes*, Br Telecom Technol Jnl, Vol 8 No3.

Walden, B.E., etc. (1977) *Effects of training on the visual recognition of consonants*,Jnl. Speech & hearing research, 20.

Welsh, W.J. & Searby, S. & Waite, J.B. (1990) *Model-based image coding*, Br Telecom Technol Jnl, Vol 8 No3, July.

Whybray, M.W. ,etc. (1990) *Videophony*, Br Telecom Technol Jnl, Vol 8 No 3,July.

Zakaria, M.F., etc. (1987) *Fast algorithm for the computation of moment invariants*, Pattern Recognition 20.