情 報 メ デ ィ ア 15－2
（1994．3．11）

視線を伴った表情とのインタラクション

内藤 剛人† 竹内 彰一‡ 所 真理雄†,‡

† 慶應義塾大学大学院 理工学研究科
‡ (株) ソニーコンピュータサイエンス研究所

概要

TV カメラから入力される画像を利用して視界内のユーザを認識し、ユーザの動きへのリアクションとして、自然な視線を伴った表情合成を行なう対話システムを開発した。このとき、計算機の前に座っているユーザとアイコンタクトを取らせようとすると、正面に座っているユーザ以外は正確にアイコンタクトを取ることができない。これはディスプレイの投影面が平面であるためである。そこで、ユーザの位置に応じて画像を歪ませることで疑似的に顔の回転を行ない、アイコンタクトを確立するアルゴリズムを考案した。また、台座に据え付けた液晶ディスプレイをモータで駆動することで、実際に首の回転をさせる装置の開発も行ない、検討を行なった。

Interactions between Human and Synthesized Facial Display with Autonomous Eye-Direction

Taketo Naito† Akikazu Takeuchi‡ Mario Tokoro†,‡

† Department of Computer Science, Keio University
3-14-1, Hiyoshi, Kohoku-ku, Yokohama, 223, Japan
‡ Sony Computer Science Laboratory Inc.
Takanawa Muse Building, 3-14-13 Higashi Gotanda
Shinagawa-ku, 141, Japan

Abstract

In this paper, we developed an interactive system that can recognize a position of the user in the computer's view, and synthesis facial displays with autonomous eye-direction as the reaction to motion of user. The user who does not stand just in front of the display cannot establish precise eye-contact with the facial image, because the surface of the screen is flat. So we devised an algorithm that rotates face virtually by deformed image according to the position of the user. Also we developed a device that consists of the stepping motor mounted with LCD, and we compare both methods.

1 はじめに

音声による計算機の操作は、計算機を意識せずに利用するためのインタフェースとして長い間研究されてきた。しかし、人と人とのコミュニケーションは、情報のチャネルが音声だけというのはむしろ特殊な場合であり、一般的には face-to-face で行なわれる。その理由は、顔は単に口から音声を発するだけでなく、同時に作り出す視線や表情によってそこに付加情報を与えているからである。また、音声や表情は互いが補い合うことで、それぞれ単独のときには得ることのできない情報を生み出すこともできるのである [1][7]。

インタフェースとしての顔を考えてみると、顔には親しみや信頼感、愛情といった情緒的な感情を抱かせる要素もある。これは、これまでの人と機械といった関係とは違う、人と計算機との新たなパートナーシップを築く手段として大いに有望である。近年のグラフィクス機能の大幅な向上により、高品質な顔画像をリアルタイムでアニメーションできるようになった。こうした顔によるインタフェースは近い将来に実用化が可能である。しかし、高度な表情合成を行なったとしても、計算機の前に座っている人を認識した上で表情を合成していなければ、精巧にできたロボットと同じで、人はその顔に対して感情移入ができないだろう。計算機が人間と同様に外界を認識し、状況に応じた適切なリアクションを返すことができて、はじめて人間らしい表情を持つことが可能となるのである。

ところで視線は、顔が発信する情報の中で表情と並んで重要な役割を果たしている。視線自身は、現在見ている物体の方向や、見ている物体までのおおよその距離といった情報しか提供しない。ところが我々はそこから、相手は現在何に興味を持っているのか、さらには現在何を考えているのかといったことまで推測することができる。また、複数人で対話をしている場合は、話しかけている人が誰かを特定したり、目配せすることで発言権をスムースに委譲したりもできる。もし計算機が自律的に視線を生成し、対話に活用することができれば、ユーザは計算機の中の顔と本当に対話している気分になれるであろう。

そこで我々は、視界内のユーザの動きに反応して、自然な視線を伴った表情を生成するようなインタラクティブシステムの開発を行なった。本稿では、まず 2 章でインタラクティブシステムの各部について説明する。今回のシステムを開発するにあたって、通常のディスプレイでは、正面以外に座っているユーザには正確な顔画像を見せることができないことが判明した。これでは、複数のユーザとそれぞれアイコンタクトを取りながら対話を行なうようなインタフェースには利用することができない。そこで 3章では、画像を歪ませることで疑似的に顔の回転を行なわせる方法と、モータに据え付けた液晶ディスプレイに顔を表示させて回転させる方法を行ない、比較する。最後に 4章で結論を述べる。

2 インタラクティブシステム

実験システムでは、グラフィクスワークステーション (SGI IRIS、以下 WS) にビデオボード (Video Lab) を介して CCD カメラが接続されている (図 1)。一般に画像処理には特殊な装置を必要とすることが多いが、本システムはソフトウェアのみで画像認識とグラフィクス生成のリアルタイム処理を可能にしている。

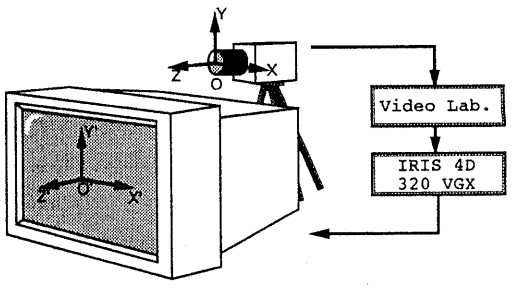

図 1: システム構成

インタラクティブシステムは、動画像処理の結果得られたユーザの位置情報をもとに目と顔の回転角の計算を行ない、視線・顔の向き・表情をリアクションモデルを基に生成し、それを表情合成モジュールに対して命令を送っている。表情合成モジュールは、今回のシステムのベースとなった表情合成システム [2] のモジュールを利用している。この表情合成システムは、モデリング、表情合成、コントローラの 3 つのモジュールから構成されている。コントローラモジュールは、表情を作るための顔面の各筋肉の収縮度データを表情合成モジュールに送る。このデータを元に表情合成モジュールは、モデリングモジュールによって二次元画像 (写真、ビデオなど) から作られた三次元顔面モデル (図 2) に対して、筋肉に基づいた変形 [3] を加えて、連続した表情アニメーションを生成する。

以下では、ユーザ位置の認識のための動画像処理、視線生成のための角度計算、リアクションモデルについて順に述べていく。

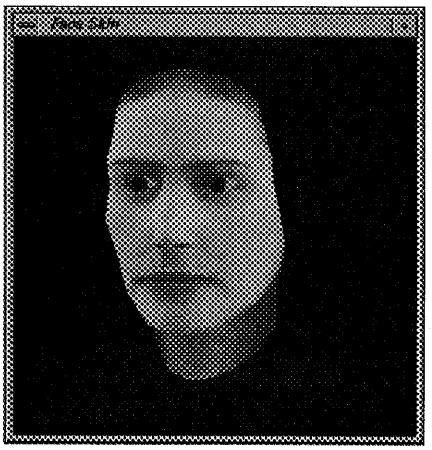

図 2: 表情の合成

2.1 動画像処理

動画像処理では、(1) 入力画像の解像度を落す解像度変換部、(2) ユーザのいる領域を抽出する差分計算部、(3) 抽出された複数の領域を識別するラベルづけ部、(4) 各領域の重心点を求める重心計算部の 4 つの処理が行なわれる。

(1) 解像度変換

カメラが撮影した画像は、解像度が 646×486 ピクセルの 24bit カラー RGB 画像として、ビデオボードを介して WS のグラフィクスバッファに直接転送される。このような高い解像度では、膨大なピクセル数のため処理に時間がかかってしまう。しかし、今回設計を行なった基本的リアクションは、ユーザまでの距離を仮定しておけば、ユーザの位置情報のみで行なうことができるので、高解像度の画像は必要としない。そこで、画像に解像度変換を施して、認識可能な範囲で解像度を落し、これを (2) 以降の処理に利用する入力画像とすることにした。

このとき、変換処理自体もできる限り計算量が少ない方が良いため、ここでは単純に 20 ライン、20 ピクセル毎にピクセルを拾っていくことにした。結果として、入力画像は元のピクセル数の 1/400 の 32×24 ピクセルにまで落すことができた。これにより、1 台の WS 上で画像処理と表情合成の両方をリアルタイムで行なうことが可能となっている。図 3は画像処理が行なわれた結果の画像である。黒い領域は、ユーザの上半身が視界に入っていることを表している。

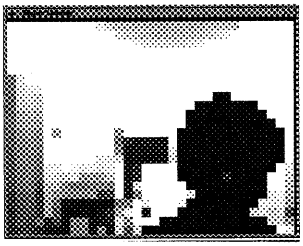

図 3: 入力画像の処理結果

実験システムが設置されている部屋は蛍光灯照明のため、この変換処理では部屋の明度変化によるノイズが画像全体に発生しやすい。また、カメラ自身によるノイズも発生する。しかし、後述する重心計算においてこれらのノイズの影響をほぼ取り除くことができるので、変換処理は今回の方法で問題がないことが分かっている。

(2) 画像の差分計算

連続して入力される画像から、動いているユーザを認識するために、まず視界内で動く物体は、ユーザのみであると仮定している。このとき、もしユーザが視界に入ると、部屋の背景画像はユーザに遮られることになり、そこの RGB 値は変化する。そこで、あらかじめユーザがいない状態の部屋を背景画像として記録しておき、時々刻々と送られてくる入力画像と背景画像のピクセルごとの差分を取ることで、以下の式のいずれかを満たすピクセルの位置には、ユーザが存在するということが分かる [4]。

$$
\begin{aligned}
|R_{video} - R_{background}| &> R_{threshold} \\
|G_{video} - G_{background}| &> G_{threshold} \\
|B_{video} - B_{background}| &> B_{threshold} \\
(0 \leq R, G, B &\leq 2^8 - 1)
\end{aligned}
$$

この式で閾値が高いほど、前述したノイズの除去効果は高いが、閾値を高くし過ぎるとユーザの存在する領域の検出が困難になる。そこで、両者のトレードオフを考慮して実験を行ない、R、G、B の各閾値を 25(2^8 の約 1 割) とした。

(3) ラベルづけ

差分計算によって抽出されたピクセルは、画像全体を走査し、隣接するピクセル同士を連結させることで、

一つの領域にまとめられる。複数のユーザが視界に入っている場合、各領域の識別も行なうために、2パスの処理を行なってラベルづけする必要がある [4]。

(4) 重心計算

視線生成の際には、両眼球が見つめる注視点を決定しなければならない。ユーザと厳密なアイコンタクトを取らせる場合には、ユーザの目を注視点とすべきであるが、ここでは処理の軽減のため、領域の重心を注視点としている。各領域 $Region_i(i=1\ldots m)$ の重心 $B_i(x_{B_i}, y_{B_i})$ は、以下の式で求められる。

$$x_{B_i} = \frac{\sum x_j}{S_i} = \frac{\sum x_j}{n}$$

$$y_{B_i} = \frac{\sum y_j}{S_i} = \frac{\sum y_j}{n}$$

$$x_j, y_j \in Region(j=1\ldots n)$$

認識精度をあげるために、差分計算の閾値はノイズを完全に除去するほど高く設定されていない。しかし、ノイズは画面全体に点在して現れるため、個々のノイズの面積はたかだか5~10程度である。そこで、重心計算の際に基準以下の面積の領域はノイズとみなし、計算対象から除外することでノイズによる影響を除去することにした。

以上の動画像処理の流れを示したのが、図4である。図では、領域の重心が黒い点で表示されている。

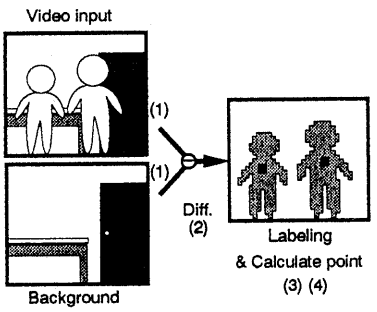

図 4: 動画像処理の流れ

2.2 視線の角度計算

視線は、眼球中心と瞳孔を結ぶ両目のそれぞれの直線が、観察する物体において交わることで形成される。視線の交差する角度は輻輳と呼ばれ、輻輳のための筋肉の緊張度は人が距離感を知覚するのに利用されている。逆に我々は、この輻輳を観察することで相手が見ている物体までの距離をおおよそ知ることができる。輻輳がないと例え視線を自分に向けられていても、自分の後ろを見ているような違和感を感じてしまう。したがって、輻輳も考慮して計算機の顔がユーザを見つめているように視線を生成しなければならない。視線を重心の所で交差させるようにするためには、重心までの距離が分かっている必要がある。しかし、今回はカメラ一台を使用し、単純な動画像処理しか行なっていないため、距離の判別までは行なえない。そこで本システムでは、ユーザまでの距離をディスプレイから50[cm] と仮定している。

はじめに、(1) 入力画像をもとにカメラから重心までの角度を計算し、(2) 目から重心までの角度に変換を行なう。

(1) カメラ座標系の角度計算

カメラの焦点から入力画像上の任意の点 (V_x, V_y) までの角度 θ_{cam} は次の式、

$$\theta_{cam_x} = \tan^{-1}\frac{V_x}{f}$$

$$\theta_{cam_y} = \tan^{-1}\frac{V_y}{f}$$

でそれぞれ求められる (図5)。ここで、f はカメラレンズの焦点距離であり、事前にキャリブレーションによって求めた値を使用している。カメラの視野角は、左右約±30°、上下約±23°となっている。

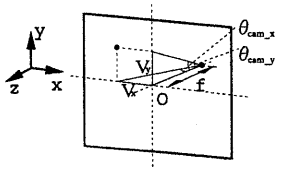

図 5: カメラ角度の算出

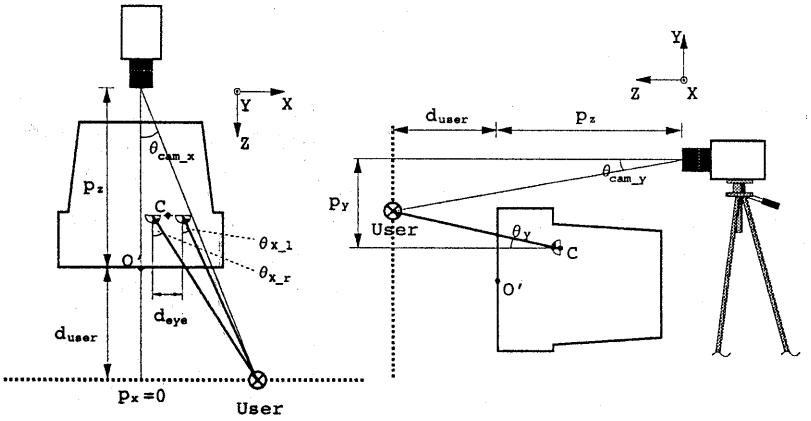

図 6: カメラ角度から目の回転角への変換

(2) 目の回転角への変換

カメラのレンズ中央からディスプレイの投影面中央まで向かう変位ベクトルを$\vec{P}(p_x,p_y,p_z)$、ディスプレイからユーザまでの距離を$d_{user}=50$[cm]、顔の眉間(両目の間)の座標を$C(c_x,c_y,c_z)$、両目の間隔をd_{eye}とする。すると、両目のx,y方向の角度はそれぞれ以下のように求まる(図6)。

$$\theta_x = \tan^{-1}\frac{(p_z+d_{user})\tan\theta_{cam_x}-c_x \pm \frac{d_{eye}}{2})}{d_{user}-c_z}$$

$$\theta_y = \tan^{-1}\frac{(p_z+d_{user})\tan\theta_{cam_y}+p_y}{d_{user}-c_z}$$

$\vec{P}(p_x,p_y,p_z)$ や d_{user}の単位は [cm] で与えられているため、角度計算の際には単位を統一する必要がある。ここでは、[cm] をディスプレイのドットピッチ(= 0.26[mm/dot]) を用いて、ドット数 [dot] に変換し、さらにウィンドウ内のドットと正規座標との比率$Ratio_x, Ratio_y$を用いて、計算機内部の正規座標系へと変換している。

2.3 リアクションモデル

動画像処理および角度計算によって与えられるユーザまでの角度を入力とし、顔の向きを含めた視線を伴う表情を出力とするような、外界の変化に反応できるリアクションモデルの構築を行なった。実験システムでは、基本的リアクションとして、以下のものが行なえる。

(1) ユーザの認識

我々は、相手が自分のことを見ているかどうかを表情から知ることができる。そして相手が自分を認識してくれて、はじめて対話を始めることができる。そこで、計算機がユーザを認識していることを表情で知らせるようにする。表情は、ユーザが自然に対話を行なえるようにするために、視界内にユーザが認識されている間、絶えず口元に微笑を浮かべるようにしている。ユーザが認識されない場合は、筋肉を全て弛緩させた無表情に戻す。こうすることで、ユーザは計算機が自分を認識しているかどうかを表情から知ることができる。

(2) 視線および顔の向きによるユーザ追跡

ユーザが動くと、その動きを視線および顔の向きで追跡する。このとき、目と顔を一緒に回転させると、目は顔に対して常に正面を向いていることになり、人形のようで不自然である。そこで、目は常に回転するようにし、現在の顔の回転角θ_{face}と目の回転角θ_xの間に10°前後の差がついた時に、顔を$\theta_x-\theta_{face}$だけ回転させるようにした。こうすることで、現在の顔の向きが大まかなユーザの位置を示し、視線はより詳細な情報を示すようになった。

(3) 瞬目

目元をより人間らしくするため、瞬目 (瞬き) を行なわせている。瞬目に関しては、心理学においてその生起原因について様々な研究が行なわれているが [5]、ここでは実際の人間を参考に、約 10 秒おきに瞬きを行なうようにしている。また、(2) で顔を回転させる際にも 3 回に 1 回の割合で瞬目を行なうようにしている。前者で眼球の湿潤性を保つための瞬目を表現し、後者では外界の知覚の結果生じた瞬目を表現している。以上 2 つのルールのみでも、人間らしい瞬目を行なうようになり、目元の表情がより豊かになった。

(4) 挨拶

対話の開始を知らせるために、視界に人が入ってきたら、その人の方向を向いて笑顔でおじぎをして挨拶を行なう。このときの笑顔と (1) の微笑は、筋肉エディタ [6] を用いて設計された表情で、実験によって求められた 20 人の笑顔の筋肉のパラメータの平均値から作られている。

3 顔の向きに関する検討

顔によるインタフェースで複数のユーザと同時に対話を行なう場合、正確にアイコンタクトを取れると対話をより円滑に進めることができる。アイコンタクトを取る場合は、視線とともに顔の向きが重要になってくる [8]。しかし、通常の二次元ディスプレイに正面を向いた顔画像を表示すると、ディスプレイの正面に座っている人には顔が正面を向いているように見えるが、このとき斜め方向から見ている人に対しても正面を向いているように見えてしまう。一方、斜め方向の人とアイコンタクトを取らせるために、顔を回転させてみると、正面の人はもちろん、斜めから見ている人とも視線を合わせることができなくなってしまう。これは、計算機内の顔面モデルがディスプレイの投影面に対して垂直に投影された二次元の画像として表示されているのが原因である。

本章では、こうした通常のディスプレイで斜め方向の人とアイコンタクトを取る方法について検討を行なう。ディスプレイを直接回転することができれば、ユーザも顔の回転を認識しやすいが、ブラウン管は重量があるため、困難である。そこで今回は、ディスプレイへ顔面モデルを投影する比率を、ユーザの位置に応じて変化させて画像を歪ませ、疑似的にディスプレイを回転させることで顔を向けさせる実験を試みた (図 7)。また、小型の液晶ディスプレイをモータに載せそこに顔を表示し、首の回転をコントローラで制御するという方法も試みた。以下で、それぞれの方法について説明する。

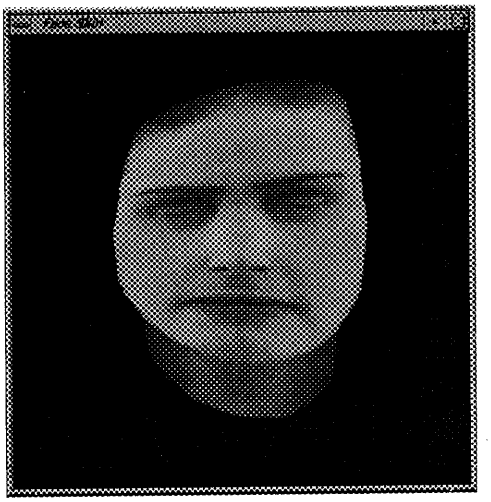

図 7: 歪ませた顔

3.1 疑似的な顔の回転

正面に座っているユーザに顔の正面を向ける場合には、図 8(a) のように、ディスプレイに垂直に投影すれば良い。もし、ユーザがθ_{view}の角度から見ている場合には、ディスプレイの投影面がユーザに対して垂直になるようにディスプレイおよび顔を疑似的に回転し (図 8(b) 点線)、そこに顔画像が垂直に投影されていると仮定する。このとき、疑似投影面に向けて垂直に投影される画像を、実際の投影面で切った断面が求める画像となる。

ディスプレイ投影面の中央を原点 O'とし、ディスプレイ内の三次元座標を (x,y,z) とすると、x 座標は、

$$x' = x - z \tan\theta_{view}$$

となる。これを、顔面モデルの座標に対して適用すれば良い。

ところが、この変形だけではある程度θ_{view}が大きくなると、ユーザに対するディスプレイの傾きの増大により、奥の部分ほど y 軸方向の長さが短く見えてしまう現象が起こる。そこで、奥の部分ほど大きくし、手

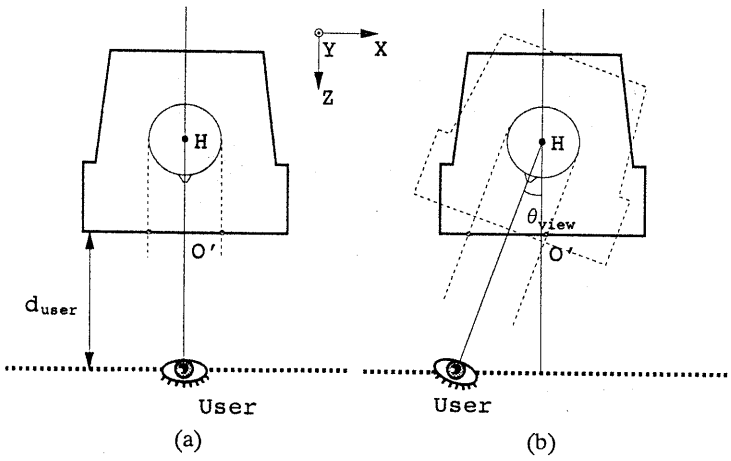

図 8: ディスプレイの疑似的な回転

前ののものほど小さくなるようにする。これは、頭部の中心 $H(H_x, H_y, Hz)$ における y 軸方向を長さの基準として、ユーザからの距離に応じて y 方向の長さを変化させることで行なわれる。その結果、y 座標は、

$$y' = y \times \frac{\sqrt{\{x+(d_{user}-H_z)\tan\theta_{view}\}^2+(d_{user}-z)^2}}{(d_{user}-H_z)\tan\theta_{view}}$$

となる。

これらの変形の結果、疑似投影面に対して垂直な位置にいるユーザのみが正しい画像を見ることができ、他のユーザは歪んだ画像を見ることになる。しかし、実際に表示してしてみたところ、疑似的に自分の方を向いているようには見えず、やはり歪んだ顔に見えてしまう。これは、正しい顔画像に見えるはずであっても、人はディスプレイの筐体やウィンドウの枠などの長方形の各辺の比率から、顔の画像を補正して見てしまい、歪みを感知してしまうからであると考えられる。したがって、標準的な計算機のディスプレイや液晶テレビなど小型のディスプレイでは本手法はあまり有効ではないが、劇場のスクリーンや大型プロジェクションディスプレイなどでは、有効であると考えられる。

3.2 ディスプレイの回転

前節の疑似的な回転とは対照的に、ここでは 4inch の液晶ディスプレイを台座に据え付け、首の役割としてステッピングモータで回転させる方法を行なう。モータの制御は、RS232C で接続されたコントローラを利用する。コントローラは、RS232C インタフェースカード、Z80CPU ボード、2 相モータドライバから構成されており、ワークステーションのシリアルポートから送られてくる命令に応じて、ディスプレイを載せたステッピングモータを回転させる。コントローラが受け付ける命令は、右回転角度、左回転角度、回転速度の 3 つのみで、これらを変換してモータに対してパルスを送る。モータは 1 パルスで 1.8°回転する。

図 9で示されるように、表情合成モジュールに顔の y 軸に関する回転角度を送らずに、代わりにモータのコントローラへ送ることで、ディスプレイの中の顔は回転せず、モータの回転が行なわれるようになる。

4 結論

表情合成システムにカメラを接続し、目としての働きを持たせることで、ユーザの動きを認識し、自然な視線を伴った表情で反応するリアクションシステムの開発を行なった。顔が自律的な視線を生成できるようになった結果、目元の表情が豊かになり、生命感を感じさせる表情を合成できるようになった。

さらに、複数のユーザとのインタラクションを考慮

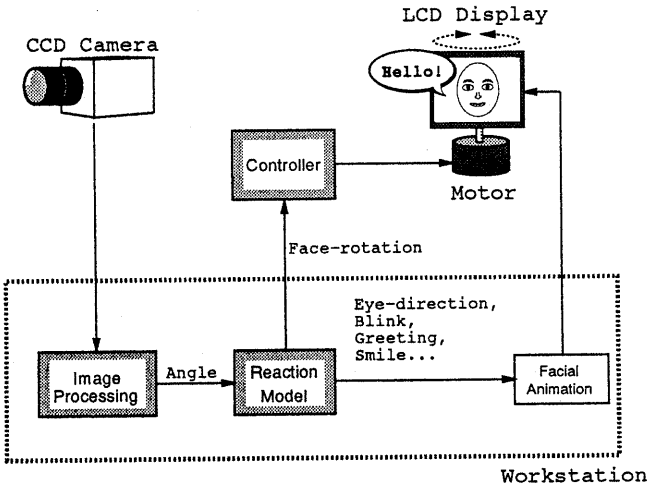

図 9: リアクションモデル・アーキテクチャ

した場合、正確にアイコンタクトを確立できるとより円滑にインタラクションを行なうことができるが、通常のディスプレイでは正確なアイコンタクトを行なうのは困難である。そこで、顔画像を歪ませることによる疑似的な顔の回転方法と、モータに据え付けた液晶ディスプレイを回転させる方法の二つを試みた。顔は多少横に横に歪んでもあまり違和感を感じないために、疑似的な回転方法では即座に現在の向きを知ることが困難であったが、モータの回転装置は実際に作動させてみると、モータの駆動音や液晶ディスプレイの向きなどで顔がどこを向いているのかを即座に知ることができ、計算機に存在感を与えることができた。

今後はリアクションモデルの拡張とともに、音声対話システムとの統合を予定している。

謝辞

本研究を行なうにあたり御支援を頂いた沼岡氏をはじめとするソニー CSL の皆様、ならびに慶應義塾大学所研究室の皆様に感謝致します。

参考文献

[1] 長尾確, 徳永健伸. 21 世紀のヒューマンコンピュータインタラクション. 情報処理, Vol. 34, No. 11, pp. 1325–1334, 1993.

[2] A. Takeuchi and S. Franks. A Rapid Face Construction Lab. Technical report, SCSL-TR-92-010, Sony Computer Science Laboratory, Inc. Tokyo, 1992.

[3] K. Waters. A Muscle Model For Animating Three-Dimensional Facial Expression. *Computer Graphics*, Vol. 22, No. 4, pp. 17–24, 1987.

[4] 谷内田正彦. ロボットビジョン. 昭晃堂, 1990.

[5] 多田英興, 山田富美雄, 福田恭介. まばたきの心理学. 北大路書房, 1991.

[6] 内藤剛人, 竹内彰一, 所真理雄. 筋肉エディタによる表情アニメーションの向上. 情報処理学会 グラフィクスと CAD シンポジウム, pp. 69–78, 1993.

[7] N. Chovil. Discourse-Oriented Facial Displays in Conversation. *Research on Language and Social Interaction*, Vol. 25, pp. 163–194, 1991.

[8] K. Morii, F. Kishino, and N. Tetsutani. Evaluation of a gaze using real-time CG eye-animation combined with eye movement detector. In *Proc. of the Fifth Int. Conf. on Human-Computer Interaction*, pp. 1103–1108, 1993.

[9] A. Takeuchi and K. Nagao. Communicative Facial Displays as a New Conversational Modality. In *Proc. INTERCHI'93*, pp. 187–193, 1993.