A two-channel training algorithm for hidden Markov model to identify visual speech elements

Foo, Say Wei; Yong, Lian; Dong, Liang

2003

Foo, S. W., Yong, L., & Dong, L. (2003). A two-channel training algorithm for hidden Markov model to identify visual speech elements. In Proceedings of the International Symposium on Circuits and Systems 2003: (pp.572-575). Singapore.

https://hdl.handle.net/10356/90658

https://doi.org/10.1109/ISCAS.2003.1206038

A TWO-CHANNEL TRAINING ALGORITHM FOR HIDDEN MARKOV MODEL TO IDENTIFY VISUAL SPEECH ELEMENTS

Say Wei Foo

School of Electrical and Electronic Engr.
Nanyang Technological University, Singapore
eswfoo@ntu.edu.sg

Yong Lian and *Liang Dong*

Department of Electrical and Computer Engr.
National University of Singapore
{eleliany, engp0564}@nus.edu.sg

ABSTRACT

A novel two-channel algorithm is proposed in this paper for discriminative training of Hidden Markov Models (HMMs). It adjusts the symbol emission coefficients of an existing HMM to maximize the separable distance between a pair of confusable training samples. The method is applied to identify the visemes of visual speech. The results indicate that the two-channel training method provides better accuracy on separating similar visemes than the conventional Baum-Welch estimation.

1. INTRODUCTION

HMM is a powerful statistical tool of analyzing time-delayed process. Because of its great adaptability and flexibility in dealing with sequential signals, HMM is heavily applied to the areas such as speech recognition, handwriting recognition, speaker identification and so on. Among the various training strategies adopted in these applications, the Baum-Welch estimation becomes popular due to its fast convergence and ease of implementation. However, this method has its deficiencies in fine discrimination. The HMM obtained from the Baum-Welch estimation is solely determined by the correct observations but does not consider the relationship between the correct observations and incorrect ones. As a result, it may be ineffective to distinguish similar observations because the scored probabilities of incorrect ones are likely to be big too. A solution to this problem is maximum mutual information (MMI)-based estimation [1]. This method increases the *a posteriori* probability of the model corresponding to the training data and thus the discriminative power is guaranteed. However, because it is difficult to realize the analytical solutions to MMI criterion function, its implementation is always tedious.

In this paper, a new strategy is proposed to improve the discriminative power of the HMMs. The symbol emission coefficients of a Maximum Likelihood (ML) HMM are partitioned into two parts (channels) for different processing. One channel is a normal HMM channel, the other is adjusted to maximize the separable distance between the correct samples and certain group of incorrect samples. In such management, the parameters of the HMM are much easier to adjust than in MMI estimation. As the expectation-maximization (EM) estimation is applied, the new HMM is trained with a few added iterations.

The two-channel training algorithm is applied to identify the basic visual speech elements in English, the visemes. Experimental results indicate that the two-channel training algorithm is more effective in distinguishing similar visemes than the Baum-Welch method.

2. TECHNICAL BACKGROUND

Assume $\{S_1, S_2, \cdots S_N\}$ is the set of states and $\{O_1, O_2, \cdots O_M\}$ is the set of observations, an *N*-state-*M*-symbol discrete HMM $\theta(\pi, A, B)$ is determined by 1.) Initial state probabilities $\pi = [\pi_i]_{N\times 1}$. 2.) State transition matrix $A = [a_{ij}]_{N\times N}$ and 3.) Symbol emission probability matrix $B = [b_{ij}]_{N\times M}$. If a *T*-length sequence $x^T = (x_1^T, x_2^T, \cdots, x_T^T)$ is the sample of class d_i, an ML HMM $\theta_i(\pi, A, B)$ can be obtained by using the Baum-Welch estimation. However, since the Baum-Welch estimation is not for discrimination purpose, if there is another sample $y^T = (y_1^T, y_2^T, \cdots, y_T^T)$ that is similar to x^T but belongs to a different class $d_j\,(j \neq i)$, the scored probability $P(y^T \mid \theta_i)$ is likely to be big too, thus θ_i cannot distinguish x^T and y^T with good credibility. To improve the discriminative power of the HMM on separating them, parameters in θ_i should be adjusted to maximize the following separable distance.

$$I(x^T, y^T, \theta) = \log P(x^T \mid \theta) - \log P(y^T \mid \theta) \qquad (1)$$

In the proposed strategy, only the coefficients in matrix *B* of θ_i are modified while *A* and π are left unchanged because: 1.) the parameters of the HMM are usually carefully initialized so that the states of the trained ML HMM are physically aligned with certain phase of a process. For example, states in speech modeling may represent different phonemes. If *A* and π are modified, such correspondence is violated. 2.) the proposed training strategy is a sub-HMM method. That is to say, the discriminative power of an HMM is improved by enhancing the discriminative ability of the individual state. Changing the state duration will greatly complicate the problem.

With the above assumption and if only the probability constraint $\sum_{j=1}^{M} b_{ij} = 1 \quad (i = 1,2,\cdots N)$ is considered, maximizing (1) is equivalent to maximizing the auxiliary function (2),

$$F(x^T, y^T, \theta, \lambda) = I(x^T, y^T, \theta) + \sum_{i=1}^{N} \lambda_i (1 - \sum_{j=1}^{M} b_{ij}) \qquad (2)$$

where λ_i is the Lagrange multiplier for the *i*-th state. Differentiating $F(x^T, y^T, \theta, \lambda)$ with respect to b_{ij} and set the result to 0, we have,

0-7803-7762-1/03/$17.00 ©2003 IEEE

$$\frac{\partial \log P(x^T \mid \theta)}{\partial b_{ij}} - \frac{\partial \log P(y^T \mid \theta)}{\partial b_{ij}} = \lambda_i \tag{3}$$

$I(x^T, y^T, \theta)$ attains the maximum value if the solutions of b_{ij} are positive. Performing the indicated differentiations by breaking the likelihoods in (3), we have, after some manipulation,

$$\frac{\partial \log P(x^T \mid \theta)}{\partial b_{ij}} = \frac{1}{b_{ij}} \sum_{\tau=1}^{T} P(s_\tau^T = S_i, x_\tau^T = O_j \mid \theta, x^T) \tag{4}$$

where $s_1, s_2, \cdots s_T$ denote the state chain.

Let $E(S_i, O_j \mid \theta, x^T) = \sum_{\tau=1}^{T} P(s_\tau^T = S_i, x_\tau^T = O_j \mid \theta, x^T)$, $E(S_i, O_j \mid \theta, y^T)$ $= \sum_{\tau=1}^{T} P(s_\tau^T = S_i, y_\tau^T = O_j \mid \theta, y^T)$, and define $D_{ij}(x^T, y^T, \theta) = E(S_i, O_j \mid \theta, x^T) - E(S_i, O_j \mid \theta, y^T)$, (4) becomes

$$\frac{D_{ij}(x^T, y^T, \theta)}{b_{ij}} = \lambda_i \quad with \quad \sum_{j=1}^{M} b_{ij} = 1 \tag{5}$$

The extreme point of $I(x^T, y^T, \theta)$ is determined by (5). However, this equation cannot be applied directly to modify b_{ij} because 1.) the numerator may be less than or equal to 0, and 2.) the unknown entry b_{ij} also exists in calculating $E(S_i, O_j \mid \theta, x^T)$ and $E(S_i, O_j \mid \theta, y^T)$. (5) just suggests a recursion strategy of optimizing b_{ij} through the expectation of $D_{ij}(x^T, y^T, \theta)$.

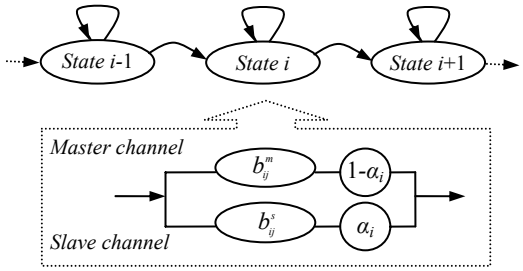

Figure 1. Structure of a two-channel HMM

To modify the parameters according to (5) and simultaneously maintain the validity of the model, a two channel structure is devised as shown in Fig. 1. The symbol emission coefficient, e.g. b_{ij} is decomposed as $b_{ij} = b_{ij}^m + b_{ij}^s$. For the i^{th} row, the coefficients are decomposed into the sum of two parameter sets, which are called the master channel and the slave channel.

$$\{b_{i1}\ b_{i2} \cdots b_{iM}\} = \underbrace{\{b_{i1}^m\ b_{i2}^m \cdots b_{iM}^m\}}_{master\ channel} + \underbrace{\{b_{i1}^s\ b_{i2}^s \cdots b_{iM}^s\}}_{slave\ channel} \tag{6}$$

The coefficient set $\{b_{i1}^m\ b_{i2}^m \cdots b_{iM}^m\}$ constitutes the master channel, which serves to maintain the validity of the HMM. For this purpose, the ML model θ_i undergoes parameter smoothing (if the estimation of certain symbol emission coefficient $b_{ij} = 0$, a small positive value, e.g. 10^{-3} is assigned to b_{ij}. As a result, the scored probability of any non-zero sequence given θ_i is greater than 0.), and b_{ij}^m's are derived from (7), where $\tilde{b}_{ij}$ is the symbol emission coefficient of the smoothed θ_i.

$$\{b_{i1}^m\ b_{i2}^m \cdots b_{iM}^m\} = (1-\alpha_i)\{\tilde{b}_{i1}\ \tilde{b}_{i2} \cdots \tilde{b}_{iM}\} \quad 0 < \alpha_i < 1 \tag{7}$$

where α_i is the credibility factor of the i^{th} state that controls the weights of the two channels.

The set $\{b_{i1}^s\ b_{i2}^s \cdots b_{iM}^s\}$ constitutes the slave channel of the HMM. In contrary to the master channel, it is the key source of the discriminative power. Its parameters $b_{21}^s\ b_{22}^s \cdots b_{2M}^s$ are modified according to (5) to maximize the separable distance between the training pair. However, if the estimation of some coefficient is less than 0, it should be replaced with a non-negative value to guarantee the validity of the HMM.

3. TRAINING STRATEGY

3.1 Parameter Initialization

The master channel coefficients b_{ij}^m's are initialized as mentioned above. For the discrete HMM discussed in this paper, the slave channel coefficients b_{ij}^s's can be initialized with random or uniform values. The selection of α_i is very flexible and largely problem-dependent. If the training pair is similar to each other, greater α_i should be set to highlight the slave channel to improve the discriminative power; otherwise, smaller α_i should be set to make $P(x^T \mid \theta)$ reasonably great. In addition, different states should have different α_i because they contribute differently to the scored probability. Taking into consideration these points, the following procedure is taken to determine α_i. Given the ML HMM θ and training pair x^T and y^T, the optimal state chain is obtained via Viterbi searching [2]. If θ_i is a left-right model and the expected duration of the i^{th} state of x^T is from t_i to $t_i+\tau_i$, $P(x^T \mid \theta)$ is then written as in (8).

$$P(x^T \mid \theta) = P(x_{t_1}, \cdots x_{t_1+\tau_1} \mid \theta) P(x_{t_2}, \cdots x_{t_2+\tau_2} \mid \theta) \cdots P(x_{t_N}, \cdots x_{t_N+\tau_N} \mid \theta) \tag{8}$$

Let $P_D(x^T, S_i, \theta) = P(x_{t_i}, \cdots x_{t_i+\tau_i} \mid \theta)$, which can be computed by (9) using the forward variables $\alpha_t^x(i) = P(x_1, \cdots, x_t, s_t = S_i \mid \theta)$ and the backward variables $\beta_t^x(i) = P(x_{t+1}, \cdots, x_T \mid s_t = S_i, \theta)$.

$$P_D(x^T, S_i, \theta) = \prod_{t=t_i}^{t_i+\tau_i} [\sum_{j=1}^{N} P(s_t = S_j) b_j(x_t)] \tag{9}$$

$P_D(y^T, S_i, \theta)$ is computed in the same way. α_i is derived by comparing the corresponding $P_D(x^T, S_i, \theta)$ and $P_D(y^T, S_i, \theta)$. If $P_D(x^T, S_i, \theta) >> P_D(y^T, S_i, \theta)$, which indicates the original coefficients are good enough to discriminate x^T and y^T, a small α_i is set to better keep the original ML configurations. If $P_D(x^T, S_i, \theta) < P_D(y^T, S_i, \theta)$ or $P_D(x^T, S_i, \theta) \approx P_D(y^T, S_i, \theta)$, which indicates the state S_i is not able to distinguish x^T and y^T well, α_i must be set big enough to ensure $P_D(x^T, S_i, \bar{\theta}) > P_D(y^T, S_i, \bar{\theta})$ under the new model $\bar{\theta}$. In practical applications, α_i can be selected in a manner based on the above conditions (which is desirable), or computed via (10).

$$\alpha_i = 1/(1+Kv) \tag{10}$$

where $v = P_D(x^T, S_i, \bar{\theta}) / P_D(y^T, S_i, \bar{\theta})$ and K is a positive constant that controls the smoothness of α_i with respect to v, whose choice is also problem-dependent. After the value of α_i is settled, it cannot be modified in the training process. If θ_i is not a left-right model, e.g. an ergodic model, the expected duration of a state is several separated slices, $P_D(x^T, S_i, \theta)$ and $P_D(y^T, S_i, \theta)$ are then computed in a similar manner by multiplying the probabilities of all the slices together.

3.2 Step 1: Partition of the Symbol Set

By using the forward variables $\alpha_t^x(i)$ and backward variables $\beta_t^x(i)$, the following probability is computed,

$$\gamma_\tau^x(i) = P(s_\tau = S_i \mid x^T, \theta) = \sum_{j=1}^{N} \frac{\alpha_\tau^x(i) a_{ij} b_j(x_{\tau+1}) \beta_{\tau+1}^x(j)}{\sum_{i=1}^{N}\sum_{k=1}^{N} \alpha_\tau^x(i) a_{ik} b_k(o_{\tau+1}) \beta_{\tau+1}^x(k)} \tag{11}$$

By counting the state, we have,

$$E(S_i, O_j \mid \theta, x^T) = \sum_{\substack{\tau=1 \\ s.t.\, x_\tau = O_j}}^{T} \gamma_\tau^x(i) \tag{12}$$

and $E(S_i, O_j \mid \theta, y^T)$ is obtained similarly. It is concluded from (5) that to increase $I(x^T, y^T, \theta)$, b_{ij} should be set in proportion to $D_{ij}(x^T, y^T, \theta)$. However, for certain symbol, e.g. O_p, the expectation $D_{ip}(x^T, y^T, \theta)$ may be less than 0. These symbols need to be isolated because the probability coefficients cannot take negative values. Using (13), the symbol set $\{O_1, O_2, \cdots O_M\}$ is partitioned into the subset $V = \{V_1, V_2, \cdots V_K\}$ and its complement set $U = \{U_1, U_2, \cdots U_{M-K}\}$.

$$\{V_1, V_2, \cdots V_K\} = \arg_{O_i}[E(S_i, O_j \mid \theta, x^T) / E(S_i, O_j \mid \theta, y^T) > T] \tag{13}$$

where $T(\geq 1)$ is the threshold with typical value 1. T can also be set with larger value if we want to decrease the size of V. It is clear that $E(S_i, V_j \mid \theta, y^T) - E(S_i, V_j \mid \theta, y^T) > 0$.

3.3 Step 2: Modification to the Slave Channel

For the i-th state, the coefficient $b_i(U_j)$ should be set as small as possible. Thus we let $b_i^s(U_j) = 0$, and so $b_i(U_j) = b_i^m(U_j)$. For the set V, the corresponding slave-channel coefficient $b_i^s(V_k)$ is computed using (14), which is derived from (5).

$$b_i^s(V_k) = P_D(S_i, V_k, x^T, y^T)(\alpha_i + \sum_{j=1}^{K} b_i^m(V_k)) - b_i^m(V_k) \quad k = 1,2,\cdots K \tag{14}$$

where $$P_D(S_i, V_k, x^T, y^T) = \frac{E(S_i, V_k \mid \theta, x^T) - E(S_i, V_k \mid \theta, x^T)}{\sum_{j=1}^{K}[E(S_i, V_k \mid \theta, x^T) - E(S_i, V_k \mid \theta, y^T)]}.$$

However, some coefficient so obtained, e.g. $b_i^s(V_l)$, may be less than 0. To prevent negative values appearing in the slave channel, the symbol V_l is transferred from V to U and $b_i^s(V_l)$ is set to 0. The coefficients of the left symbols in V are re-evaluated with (14) until all the $b_i^s(V_k)$'s are greater than 0. The condition $b_i^s(V_l) < 0$ usually happens at the first several epochs of training and it is unfavorable for convergence because it leaves steep jump in the surface of $I(x^T, y^T, \theta)$. To relieve this problem, a greater threshold T in (13) should be set while obtaining V. The process will then concentrate on the fewer dominant symbols of x^T.

The two steps described above constitute a training epoch. Optimization is done through iteratively calling them. The stop condition is associated with the variance of $I(x^T, y^T, \theta)$. After each epoch, the new separable distance $I(x^T, y^T, \bar{\theta})$ is calculated and compared with that of the last epoch. If $I(x^T, y^T, \bar{\theta})$ does not change much, e.g. less than a predefined threshold, the training stops and the target model is obtained.

3.4 Discussion on the training strategy

The two-channel HMM obtained acts like boundary function as shown in Fig. 2. For example, $\theta(1,2)$ is trained to distinguish the samples of Class 1 and Class 2. Such modeling method is specially tailored for the target class and its "surroundings" and thus is more accurate than the ML models.

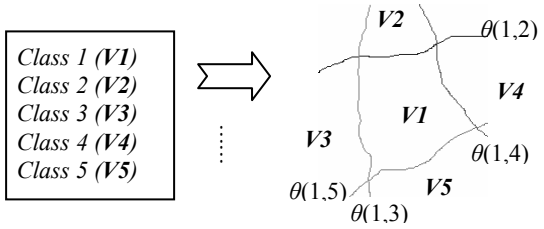

Figure 2. Class boundaries formed by two-channel HMMs

The proposed method is applicable to similar training sequences that their corresponding state durations are comparable. As it is meaningless to compare the expected symbol appearances $E(S_i, O_j \mid \theta, x^T)$ and $E(S_i, O_j \mid \theta, y^T)$ if the durations of S_i are very different in x^T and y^T. To make the training strategy complete, the following validation procedure is added. After each training epoch, the state durations $E(S_i \mid \theta, x^T)$ and $E(S_i \mid \theta, y^T)$ are computed by (15).

$$E(S_i \mid \theta, x^T) = \sum_{\tau=1}^{T} \frac{\alpha_\tau^x(i)\beta_\tau^x(i)}{\sum_{i=1}^{N} \alpha_\tau^x(i)\beta_\tau^x(i)} \quad (i = 1,2,\cdots N) \tag{15}$$

If $E(S_i \mid \theta, x^T) \approx E(S_i \mid \theta, y^T)$, e.g. $1.2E(S_i \mid \theta, y^T) > E(S_i \mid \theta, x^T) > 0.8E(S_i \mid \theta, y^T)$, the training continues; otherwise, the training terminates. If the $I(x^T, y^T, \theta)$ obtained does not meet the requirement, a new ML model or smaller α_i should be adopted.

The convergence of the training strategy is guaranteed by the Lagrange multiplier algorithm and Expectation-Maximization (EM) algorithm [3]. It can be proved that the training equation

(5) gives zero points in $\partial F / \partial \lambda$ and $\partial^2 F / \partial \lambda^2$ is negative [4]. Thus the algorithm guarantees the convergence of the training process.

The improvement to the separable distance of the proposed method is given in (16),

$$I(x^T, y^T, \bar{\theta}) \le -T \log(1-\alpha_{\max}) + I(x^T, y^T, \theta) \qquad (16)$$

where $\alpha_{\max} = \max(\alpha_1, \alpha_2, \cdots \alpha_N)$. The improvement is associated with the resemblance between x^T and y^T and the setting of the credibility factors.

The proposed method is also extended to the training sequences with different length. Given training pair x^{T_x} with length T_x and y^{T_y} with length T_y, the training equations (5) are transformed to (17), in which a linear entry T_x/T_y is adopted to normalize the state duration of y^T.

$$\frac{\sum_{\tau=1}^{T_x} P(s_\tau^{T_x} = S_i, x_\tau^{T_x} = O_j \mid \theta, x^{T_x}) - \frac{T_x}{T_y} \sum_{\tau=1}^{T_y} P(s_\tau^{T_y} = S_i, y_\tau^{T_y} = O_j \mid \theta, y^{T_y})}{b_{ij}} = \lambda_i \quad (17)$$

In multiple observations case, for example, the training pair are two labeled sample sets: $X = \{x^{(1)}, x^{(2)}, \cdots x^{(k)}\}$ and $Y = \{y^{(1)}, y^{(2)}, \cdots y^{(l)}\}$, in which $x^{(i)}$ and $y^{(i)}$ are all drawn independently, the training equation then becomes,

$$\frac{\frac{1}{k}\sum_{m=1}^{k} E(S_i, O_j \mid \theta, x^{(m)}) - \frac{1}{l}\sum_{n=1}^{l} E(S_i, O_j \mid \theta, y^{(n)})}{b_{ij}} = \lambda_i \qquad (18)$$

4. APPLICATION ON LIP READING

Viseme recognition is a good example of the application of the two-channel training algorithm. Most visemes are dynamically similar with one another because they experience the same three-phase process during production: starting from closed mouth, peaking at half-opened mouth and ending with closed mouth. By using the two-channel training, the minor difference between the visemes can be amplified via the slave channel.

3 11 11 74 74 75 75 5 5 3

Figure 3. Encoding the images using the code words

In our experiments, the image sequences that indicate viseme production are encoded with 128 code words as shown in Fig. 3. The code sequence is input to a two-layer recognition system as shown in Fig. 4. In the first layer, θ_1, θ_2 …θ_N are ML HMMs for coarse identification. After a preliminary decision is made, the vector sequence is input to a number of two-channel HMMs for fine identification. The finial decision is made by assessing the majority of the sub-decisions.

The classification errors of the two-channel HMMs are listed in Table 1 and are compared with those of the ML HMMs (It is assumed that each phoneme is associated with a viseme). For the 100 test samples of each viseme used in the experiment, the two-channel HMM usually gives a much smaller classification error than the ML HMMs. As a result, the proposed training method excels the traditional Baum-Welch estimation in identifying the confusable visemes.

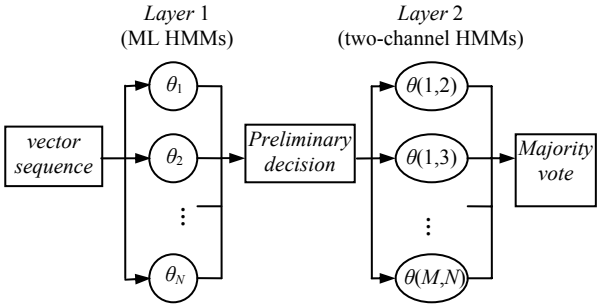

Figure 4. Block diagram of the two-layer viseme recognition system

Table 1. Classification error of the ML HMMs ε_1 and two-channel HMMs ε_2

Viseme	ε_1	ε_2	Viseme	ε_1	ε_2
/a:/	64%	12%	/ai/	59%	39%
/ei/	46%	22%	/i/	52%	31%
/au/	31%	18%	/eu/	26%	15%
/o/	47%	28%	/oi/	37%	9%
/th/	18%	17%	/sh/	20%	12%
/p/	21%	20%	/m/	32%	31%

5. CONCLUSION

A two-channel training algorithm for HMM is proposed in this paper. It is a handy method of improving the discriminative power of an existing ML HMM. The method is applied to viseme recognition and good results are obtained. This method can also be applied to sequence recognition problems that need fine discrimination, for example, speech recognition and speaker identification.

6. REFERENCES

[1] L. R Bahl, P. F. Brown, P. V. de Souza and R. L. Mercer, "Maximum mutual information estimation of hidden Markov model parameters for speech recognition," *Proc. ICASSP '86*, pp. 49-52, Apr. 1986

[2] L. R. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition", *Proc. IEEE*, pp. 257-286, Vol. 77, No. 2, Feb. 1989

[3] C. F. J. Wu. "On the convergence properties of the EM algorithm." *Annuals of Statistics*, Vol. 11, pp. 95-103, 1983

[4] X. Li, M. Parizeau and R. Plamondon, "Hidden Markov Model Multiple Observation Training", *Technical Report EPM/RT-99/16*, Nov. 1999

[5] J. K, Baker, "The Dragon System – An Overview," *IEEE Trans. on Acoustics, Speech and Signal Processing*, Vol 23, pp. 24-29, Feb. 1975

[6] M. Tekalp and J. Ostermann, "Face and 2-D mesh animation in MPEG-4," *Image Communication J.* Aug. 1999