(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
10 May 2002 (10.05.2002)

PCT

(10) International Publication Number
WO 02/37302 A1

(51) International Patent Classification[7]: G06F 15/18, G06T 1/00, G06K 9/66

(21) International Application Number: PCT/SE01/02285

(22) International Filing Date: 19 October 2001 (19.10.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
0003975-0 31 October 2000 (31.10.2000) SE

(71) Applicant and
(72) Inventor: GRANLUND, Gösta [SE/SE]; Valhallagatan 16, S-582 43 Linköping (SE).

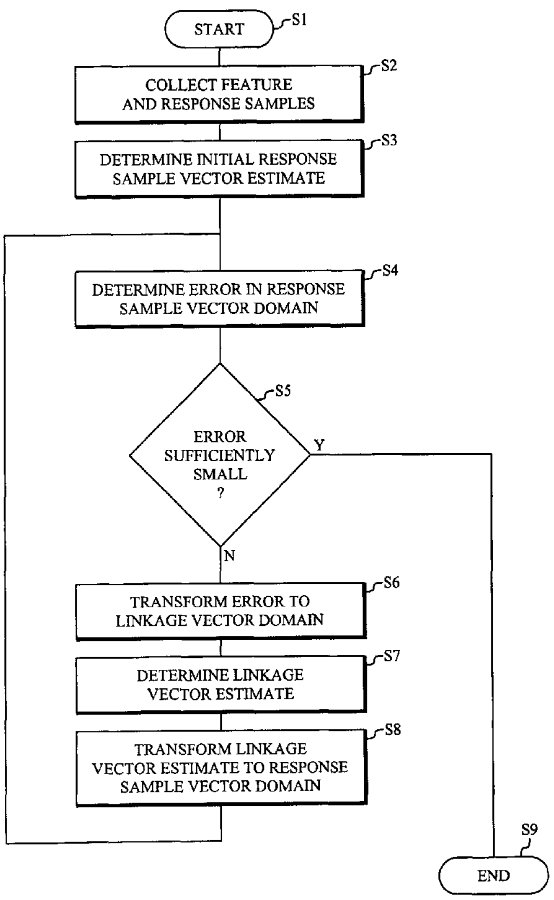
(74) Agent: HEDBERG, Åke; Aros Patent AB, P.O. Box 1544, S-751 45 Uppsala (SE).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: TRAINING OF ASSOCIATIVE NETWORKS



(57) Abstract: An artificial vision system traning method starts by forming (S2) a feature matrix including feature sample vectors and a corresponding response sample vector including response sample scalars. The method then uses an iterative procedure to determine a linkage vector linking the response sample vector to the feature matrix. This iterative procedure includes the steps: determining (S4) a response sample vector error estimate in the response sample vector domain; transforming (S6) the response sample vector error estimate into a corresponding linkage vector error estimate in the linkage vector domain; determining (S7) a linkage vector estimate in the linkage vector domain by using the linkage vector error estimate; transforming (S8) the linkage vector estimate into a corresponding response sample vector estimate in the response sample vector domain. These steps are repeated until (S5) the response sample vector error estimate is sufficiently small.

WO 02/37302 A1

Published:
— *with international search report*

TRAINING OF ASSOCIATIVE NETWORKS

TECHNICAL FIELD

The present invention relates to training of associative networks, and particularly to training of artificial vision systems or percept-response systems.

BACKGROUND

Reference [1] describes a percept-response system based on channel representation of information. A link matrix **C** links a feature vector **a**, which has been formed from a measured percept (column) vector **x**, to a response (column) vector **u** using the matrix equation:

$$\mathbf{u} = \mathbf{C}\,\mathbf{a} \tag{1}$$

A fundamental consideration in these systems is system training, i.e. how to determine the linkage matrix **C**. In [1] this is accomplished by collecting different training sample pairs of feature vectors $\mathbf{a}^i$ and response vectors $\mathbf{u}^i$. Since each pair should be linked by the same linkage matrix **C**, the following set of equations is obtained:

$$\mathbf{U} = \begin{pmatrix} u_1^1 & u_1^2 & \cdots & u_1^N \\ u_2^1 & u_2^2 & \cdots & u_2^N \\ \vdots & \vdots & \vdots & \vdots \\ u_K^1 & u_K^2 & \cdots & u_K^N \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1H} \\ c_{21} & c_{22} & \cdots & c_{2H} \\ \vdots & \vdots & \vdots & \vdots \\ c_{K1} & c_{K2} & \cdots & c_{KH} \end{pmatrix} \begin{pmatrix} a_1^1 & a_1^2 & \cdots & a_1^N \\ a_2^1 & a_2^2 & \cdots & a_2^N \\ \vdots & \vdots & \vdots & \vdots \\ a_H^1 & a_H^2 & \cdots & a_H^N \end{pmatrix} = \mathbf{C}\,\mathbf{A} \tag{2}$$

where N denotes the number of training samples or the length of the training sequence and **A** is denoted a feature matrix. These equations may be solved by conventional approximate methods (typically methods that minimize mean squared errors) to determine the linkage matrix **C** (see [2]). However, a drawback of these approximate methods is that they restrict the complexities of associative networks to an order of thousands of features and thousands of samples, which is not enough for many systems.

SUMMARY

An object of the present invention is a more efficient training procedure that allows much larger associative networks.

This object is achieved in accordance with the attached claims.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention, together with further objects and advantages thereof, may best be understood by making reference to the following description taken together with the accompanying drawings, in which:

Fig. 1 is a flow chart illustrating an exemplary embodiment of the training method in accordance with the present invention; and

Fig. 2 is a diagram illustrating the structure of a training system in accordance with the present invention.

DETAILED DESCRIPTION

Studying the structure of equation (2) reveals that each row of $\mathbf{U}$ actually requires knowledge of only the corresponding row of $\mathbf{C}$. For example, row k of $\mathbf{U}$, which is denoted $\mathbf{u}_k$, requires knowledge of only row k of $\mathbf{C}$, which is denoted $\mathbf{c}_k$. This can be seen from the explicit equation:

$$\mathbf{u}_k=\begin{pmatrix}\vdots & \vdots & \cdots & \vdots\\ u_k^1 & u_k^2 & \cdots & u_k^N\\ \vdots & \vdots & \vdots & \vdots\\ \vdots & \vdots & \cdots & \vdots\end{pmatrix}=\begin{pmatrix}\vdots & \vdots & \cdots & \vdots\\ c_{k1} & c_{k2} & \cdots & c_{kH}\\ \vdots & \vdots & \vdots & \vdots\\ \vdots & \vdots & \cdots & \vdots\end{pmatrix}\begin{pmatrix}a_1^1 & a_1^2 & \cdots & a_1^N\\ a_2^1 & a_2^2 & \cdots & a_2^N\\ \vdots & \vdots & \vdots & \vdots\\ a_H^1 & a_H^2 & \cdots & a_H^N\end{pmatrix}=\mathbf{c}_k\,\mathbf{A} \qquad (3)$$

Thus, equation (1) may be solved by independently determining each row of linkage matrix **C**. Furthermore, it has been shown in [1] that it is possible to represent a response state either in scalar representation or channel (vector) representation, and that it is possible to transform a scalar quantity into a vector quantity, or vice versa. Thus, it is possible to transform each column vector of response matrix **U** into a scalar, thereby obtaining a response sample (row) vector **u** containing these scalars as components. This response sample vector **u** will be linked to feature matrix **A** in accordance with:

$$\mathbf{u}=\begin{pmatrix}u^1 & u^2 & \cdots & u^N\end{pmatrix}=\begin{pmatrix}c_1 & c_2 & \cdots & c_H\end{pmatrix}\begin{pmatrix}a_1^1 & a_1^2 & \cdots & a_1^N\\ a_2^1 & a_2^2 & \cdots & a_2^N\\ \vdots & \vdots & \vdots & \vdots\\ a_H^1 & a_H^2 & \cdots & a_H^N\end{pmatrix}=\mathbf{c}\,\mathbf{A} \qquad (4)$$

Thus, since equations (3) and (4) have the same structure, it is appreciated that the fundamental problem is to solve an equation having the form:

$$\mathbf{u}=\mathbf{c}\,\mathbf{A} \qquad (5)$$

where **u** and **A** are known, while **c** is to be determined.

A problem with equation (5) is that response sample vector **u** and linkage vector **c** typically lie in different vector spaces or domains, since feature matrix **A** typically is rectangular and not square (H is generally not equal to N in equation (4)). Thus, feature matrix **A** has no natural inverse. In accordance with the present invention an iterative method is used to determine **c** from **A** and **u**. Since **u** is known, a current estimate $\hat{\mathbf{u}}(i)$ of response sample vector **u** is formed in the response domain, and the error $\Delta\mathbf{u}=\hat{\mathbf{u}}(i)-\mathbf{u}$ is transformed into a corresponding linkage vector error $\Delta\mathbf{c}$ in the linkage vector domain using the transpose of feature matrix **A**. This linkage vector error is subtracted from a current linkage vector estimate $\hat{\mathbf{c}}(i)$ to form an updated linkage vector estimate $\hat{\mathbf{c}}(i+1)$. This updated estimate is transformed back to the response sample vector domain using feature matrix **A**, thereby forming an updated response sample vector estimate $\hat{\mathbf{u}}(i+1)$. Thus, the iterative steps of this process may be written as:

$$\begin{aligned} \hat{\mathbf{c}}(i+1) &= \hat{\mathbf{c}}(i) - \overbrace{\underbrace{(\hat{\mathbf{u}}(i)-\mathbf{u})}_{\Delta\mathbf{u}}\cdot \mathbf{A}^T}^{\Delta\mathbf{c}} \\ \hat{\mathbf{u}}(i+1) &= \hat{\mathbf{c}}(i+1)\cdot\mathbf{A} \end{aligned} \qquad (6)$$

This procedure is illustrated in the flow chart of fig. 1. The procedure starts in step S1. Step S2 collects feature and response samples. Step S3 determines an initial response sample vector estimate, typically the zero vector. Step S4 determines the error in the response sample vector domain. Step S5 tests whether this error is sufficiently small. If not, the procedure proceeds to step S6, in which the error is transformed to the linkage vector domain. Step S7 determines a linkage vector estimate using the transformed error. Step S8 transforms this estimate back to the response sample vector domain. Thereafter the procedure loops back to step S4. If the error is sufficiently small in step S5, the procedure ends in step S9.

Fig. 2 illustrates an exemplary structure of a training system suitable to perform the described method. An response domain error is formed in an adder 10 by subtracting the actual response sample vector **u** from its corresponding estimate $\hat{\mathbf{u}}$. The response domain error is forwarded to a transformation or sampling block 12 (the transformation of the error by $\mathbf{A}^T$ may be viewed as a form of sampling). This block may also perform a normalization, a process that will be further described below. The resulting linkage vector domain error is subtracted from a current linkage vector estimate in an adder 14. This current linkage vector estimate is stored and has been delayed in a delay and storage block 16. The updated linkage vector estimate is forwarded to a transformation or reconstruction block 18, which produces an updated response sample vector estimate. This block may also perform a normalization, a process that will be further described below. The updated linkage vector estimate is also forwarded to delay and storage block 16. Both transformation blocks 12, 18 base their transformations on feature matrix **A**. Typically the blocks of fig. 2 are implemented by one or several micro processors or micro/signal processor combinations and corresponding software. They may, however, also be implemented by one or several ASICs (application specific integrated circuits).

An alternative to (6) is to change the order in the iteration in accordance with:

$$\begin{aligned}\hat{\mathbf{u}}(i+1) &= \hat{\mathbf{c}}(i)\cdot\mathbf{A}\\ \hat{\mathbf{c}}(i+1) &= \hat{\mathbf{c}}(i)-\left(\hat{\mathbf{u}}(i+1)-\mathbf{u}\right)\cdot\mathbf{A}^T\end{aligned}\qquad(7)$$

In a preferred embodiment of the present invention feature matrix A, its transpose or both are normalized. For example, in a mixed normalization embodiment the matrix $\mathbf{A}^{\mathrm{T}}$ in the linkage vector equation is feature normalized by the diagonal normalization matrix $\mathbf{N}^{\mathrm{F}}$ defined as:

$$\mathbf{N}^F=\begin{pmatrix}1/\sum_{n=1}^{N}a_1^n & 0 & \cdots & 0\\ 0 & 1/\sum_{n=1}^{N}a_2^n & 0 & \vdots\\ \vdots & 0 & \ddots & 0\\ 0 & \cdots & 0 & 1/\sum_{n=1}^{N}a_H^n\end{pmatrix}\qquad(8)$$

and the matrix **A** in the response vector equation is feature normalized by the diagonal normalization matrix $\mathbf{N}^{\mathrm{S}}$ defined as:

$$\mathbf{N}^S = \begin{pmatrix} 1/\sum_{h=1}^{H} a_h^1 & 0 & \cdots & 0 \\ 0 & 1/\sum_{h=1}^{H} a_h^2 & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & 1/\sum_{h=1}^{H} a_h^N \end{pmatrix} \tag{9}$$

Thus, in this mixed embodiment the feature normalization factors are obtained as the inverted values of the row sums of feature matrix **A**, while the sample normalization factors are obtained as the inverted values of the column sums of **A**. With this normalization (6) and (7) may be rewritten as:

$$\begin{aligned} \hat{\mathbf{c}}(i+1) &= \hat{\mathbf{c}}(i) - \left(\hat{\mathbf{u}}(i) - \mathbf{u}\right) \cdot \mathbf{N}^F \cdot \mathbf{A}^T \\ \hat{\mathbf{u}}(i+1) &= \hat{\mathbf{c}}(i+1) \cdot \mathbf{N}^S \cdot \mathbf{A} \end{aligned} \tag{10}$$

and

$$\begin{aligned} \hat{\mathbf{u}}(i+1) &= \hat{\mathbf{c}}(i) \cdot \mathbf{N}^S \cdot \mathbf{A} \\ \hat{\mathbf{c}}(i+1) &= \hat{\mathbf{c}}(i) - \left(\hat{\mathbf{u}}(i+1) - \mathbf{u}\right) \cdot \mathbf{N}^F \cdot \mathbf{A}^T \end{aligned} \tag{11}$$

respectively. As a further illustration Appendix A includes an explicit MATLAB® code implementation of (10).

Another possibility involves only feature normalizing $\mathbf{A}^T$ and retaining **A** without normalization in (6) and (7). In this case a suitable normalization matrix $\mathbf{N}^F$ is given by:

$$\mathbf{N}^F = \begin{pmatrix} 1/\sum_{n=1}^{N}\left(\sum_{h=1}^{H} a_h^n\right)a_1^n & 0 & \cdots & 0 \\ 0 & 1/\sum_{n=1}^{N}\left(\sum_{h=1}^{H} a_h^n\right)a_2^n & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & 1/\sum_{n=1}^{N}\left(\sum_{h=1}^{H} a_h^n\right)a_H^n \end{pmatrix} \quad (12)$$

With this normalization (6) and (7) may be rewritten as:

$$\begin{aligned} \hat{\mathbf{c}}(i+1) &= \hat{\mathbf{c}}(i) - \left(\hat{\mathbf{u}}(i) - \mathbf{u}\right)\cdot \mathbf{N}^F \cdot \mathbf{A}^T \\ \hat{\mathbf{u}}(i+1) &= \hat{\mathbf{c}}(i+1)\cdot \mathbf{A} \end{aligned} \quad (13)$$

and

$$\begin{aligned} \hat{\mathbf{u}}(i+1) &= \hat{\mathbf{c}}(i)\cdot \mathbf{A} \\ \hat{\mathbf{c}}(i+1) &= \hat{\mathbf{c}}(i) - \left(\hat{\mathbf{u}}(i+1) - \mathbf{u}\right)\cdot \mathbf{N}^F \cdot \mathbf{A}^T \end{aligned} \quad (14)$$

respectively. As a further illustration Appendix B includes an explicit MATLAB® code implementation of (13).

Still another possibility involves only sample normalizing **A** and retaining $\mathbf{A}^{\mathrm{T}}$ without normalization in (6) and (7). In this case a suitable normalization matrix $\mathbf{N}^{\mathrm{S}}$ is given by:

$$\mathbf{N}^S = \begin{pmatrix} 1/\sum_{h=1}^{H}\left(\sum_{n=1}^{N} a_h^n\right)a_h^1 & 0 & \cdots & 0 \\ 0 & 1/\sum_{h=1}^{H}\left(\sum_{n=1}^{N} a_h^n\right)a_h^2 & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & 1/\sum_{h=1}^{H}\left(\sum_{n=1}^{N} a_h^n\right)a_h^N \end{pmatrix} \quad (15)$$

With this normalization (6) and (7) may be rewritten as:

$$\begin{aligned}\hat{\mathbf{c}}(i+1) &= \hat{\mathbf{c}}(i) - (\hat{\mathbf{u}}(i) - \mathbf{u}) \cdot \mathbf{A}^T \\ \hat{\mathbf{u}}(i+1) &= \hat{\mathbf{c}}(i+1) \cdot \mathbf{N}^S \cdot \mathbf{A}\end{aligned} \tag{16}$$

and

$$\begin{aligned}\hat{\mathbf{u}}(i+1) &= \hat{\mathbf{c}}(i) \cdot \mathbf{N}^S \cdot \mathbf{A} \\ \hat{\mathbf{c}}(i+1) &= \hat{\mathbf{c}}(i) - (\hat{\mathbf{u}}(i+1) - \mathbf{u}) \cdot \mathbf{A}^T\end{aligned} \tag{17}$$

respectively. As a further illustration Appendix C includes an explicit MATLAB® implementation of (16).

In the description above the normalization has been expressed in matrix form. However, since these matrices are diagonal matrices, it is possible to write the iteration equations in an equivalent mathematical form that expresses the normalizations as vectors. For example, (10) may be rewritten as:

$$\begin{aligned}\hat{\mathbf{c}}(i+1) &= \hat{\mathbf{c}}(i) - \mathbf{N}^F \otimes \left((\hat{\mathbf{u}}(i) - \mathbf{u}) \cdot \mathbf{A}^T\right) \\ \hat{\mathbf{u}}(i+1) &= \mathbf{N}^S \otimes (\hat{\mathbf{c}}(i+1) \cdot \mathbf{A})\end{aligned} \tag{18}$$

where $\mathbf{N}^F$ and $\mathbf{N}^S$ now are row vectors defined by the diagonal elements of the corresponding matrices.

An essential feature of the present invention is the fact that feature matrix **A** only contains non-negative elements. It is possible to show that the gradient of the error function Δu_n is directly related to the elements of feature matrix **A**. A straightforward derivation gives:

$$\frac{\partial \Delta u_n}{\partial c_h} = a_h^n \tag{19}$$

Thus, it is appreciated that the gradient will also only contain non-negative values. This implies that it is not necessary to test the sign of the gradient.

An increase of the value of a linkage vector component c_h will move the error in a positive direction or not affect it at all. This feature is the basis for the fast iterative procedure in accordance with the present invention. A closer examination of the underlying problem reveals that the nonzero elements of **A** do not necessarily have to be positive. What is required is that they have a consistent sign (they are either all positive or all negative). Similar comments apply to **u** and **c**.

In the description above the entire feature matrix **A** is involved in the iteration. In an approximate embodiment, feature matrix **A** may be replaced by an approximation in which only the maximum value in each row is retained. This approximation may be used either throughout all equations, in one of the equations or only in selected occurrences of feature matrix **A** in the equations. An example of an approximate mixed normalization embodiment corresponding to equation (6) is given by the MATLAB® implementation in Appendix D. In this example the approximation is used in the first row of equation (6). The advantage of such an approximate method is that it is very fast, since only the maximum element in each row is retained, while the rest of the elements are approximated by zeros. After normalization, the resulting normalized matrix will only contain ones and zeros. This means that a computationally complex matrix multiplication can be replaced by a simple reshuffling of error components in error vector $\Delta\mathbf{u}$.

In a similar approximation it is also possible to approximate feature matrix **A** with a matrix in which only the maximum value of each column (sample vector) is retained. In a mixed normalization it is also possible to use both approximations, i.e. to use approximate both feature and sample normalization.

There are several possible choices of stopping criteria for the iterative training process described above.

One possibility is to use the average of the absolute value of the components of $\Delta\mathbf{u}$, i.e.:

$$\frac{1}{N}\sum_{n=1}^{N}\left|\hat{u}_n - u_n\right| \tag{20}$$

The iteration is repeated as long as a threshold *epsilon* is exceeded.

An alternative is to use the maximum error component of $\Delta\mathbf{u}$, i.e.:

$$\max_n\left(\left|\hat{u}_n - u_n\right|\right) \tag{21}$$

If large errors are considered more detrimental than smaller errors, the squared error can be used, i.e.:

$$\sum_{n=1}^{N}\left|\hat{u}_n - u_n\right|^2 \tag{22}$$

The above described stopping criteria are based on an absolute scalar error estimate. However, relative estimates are also possible. As an example, the estimate:

$$\frac{\sum_{n=1}^{N}\left|\hat{u}_n - u_n\right|^2}{\sum_{n=1}^{N}u_n^2} \tag{23}$$

has been used in the code in the appendices.

As an alternative, the iterations may be stopped after a predetermined number of iterations. A combination is also possible, i.e. if the error is not sufficiently small after a predetermined number of iterations the procedure is stopped. This may happen, for example, when some error components of the error vector remain large even after many iterations. In such a case the components of linkage vector **c** that have converged may still be of interest.

In many cases some elements of linkage vector estimate $\hat{\mathbf{c}}$ approach the value zero during the iterative process. The result of this is that the corresponding rows of feature matrix **A** will be ignored and will not be linked to the response vector. In accordance with a procedure denoted "compaction", this feature may be used to remove the zero element in the linkage vector estimate (storing its position) and its corresponding row in the feature matrix. A new normalization may then be performed on the compacted feature matrix, whereupon the iterative procedure is re-entered. This compaction may be performed each time a linkage vector element approaches zero, preferably when it falls below a predetermined limit near zero. Since the position of the removed values is stored, the complete linkage vector can be restored when the non-zero elements have converged.

As has been noted above, equation (1) may be solved by independently determining each row of linkage matrix **C**. However, the described iterative procedure may be used also for the full matrices. As an example, (6) may be rewritten as:

$$\begin{aligned} \hat{\mathbf{C}}(i+1) &= \hat{\mathbf{C}}(i) - \overbrace{\underbrace{\left(\hat{\mathbf{U}}(i) - \mathbf{U}\right)}_{\Delta\mathbf{U}} \cdot \mathbf{A}^T}^{\Delta\mathbf{C}} \\ \hat{\mathbf{U}}(i+1) &= \hat{\mathbf{C}}(i+1) \cdot \mathbf{A} \end{aligned} \qquad (24)$$

for the matrix case.

Since feature matrix **A** generally is a sparse matrix (most of the matrix elements are 0), it is preferable to implement the above described procedures in a computational system that supports sparse matrix operations. An example of such a system is MATLAB® by MathWorks Inc. This will reduce the storage requirements, since only non-zero elements are explicitly stored, and will also speed up computation, since multiplications and additions are only performed for non-zero elements.

In the description above the invention has been described with reference to an artificial vision system. However, the same principles may be applied to any percept-response system or associative network in which a feature vector including only non-negative elements is mapped onto a response vector (possibly including only 1 element) including only non-negative elements. Examples are sound processing systems and control systems based on changes in sensor variables, such as temperature, pressure, position, velocity, etc.

It will be understood by those skilled in the art that various modifications and changes may be made to the present invention without departure from the scope thereof, which is defined by the appended claims.

APPENDIX A

In MATLAB® notation the mixed normalization procedure may be written as:

```
Ns=sum(A);                  %Sample norms of A
Asn=(diag(1./Ns)*A')';      %Sample normalize A
Nf=sum(A');                 %Feature norms of A
Afn=diag(1./Nf)*A;          %Feature normalize A
epsilon=0.05;               %Desired relative accuracy
c_hat=0;                    %Initial estimate of c=zero vector
u_hat=0;                    %Initial estimate of u=zero vector

while norm(u_hat-u)/norm(u)>epsilon
    c_hat=c_hat-(u_hat-u)*Afn';   %Update estimate of c
    u_hat=c_hat*Asn;              %Update estimate of u
end;
```

Here "." denotes elementwise operations and "'" denotes transpose.

APPENDIX B

In MATLAB® notation the feature domain normalization procedure may be written as:

```
Nf=sum(A)*A';               %Feature norms of A
Afn=diag(1./Nf)*A;          %Feature normalize A
epsilon=0.05;               %Desired relative accuracy
c_hat=0;                    %Initial estimate of c=zero vector
u_hat=0;                    %Initial estimate of u=zero vector

while norm(u_hat-u)/norm(u)>epsilon
    c_hat=c_hat-(u_hat-u)*Afn';   %Update estimate of c
    u_hat=c_hat*A;                %Update estimate of u
end;
```

APPENDIX C

In MATLAB® notation the sample domain normalization procedure may be written as:

```
Ns=sum(A')*A;               %Sample norms of A
Asn=(diag(1./Ns)*A')';      %Sample normalize A
epsilon=0.05;               %Desired relative accuracy
c_hat=0;                    %Initial estimate of c=zero vector
u_hat=0;                    %Initial estimate of u=zero vector

while norm(u_hat-u)/norm(u)>epsilon
     c_hat=c_hat-(u_hat-u)*A';      %Update estimate of c
     u_hat=c_hat*Asn;               %Update estimate of u
end;
```

APPENDIX D

In MATLAB® notation the mixed approximate normalization procedure may be written as:

```
[ra ca]=size(A);            %Determine number of row and columns
[av ap]=max(A');            %Find maxima of feature functions
Ns=av*A;                    %Sample domain norms
Asn=(diag(1./Ns)*A')';      %Sample normalize A
epsilon=0.05;               %Desired relative accuracy
c_hat=0;                    %Initial estimate of c=zero vector
u_hat=0;                    %Initial estimate of u=zero vector

while norm(u_hat-u)/norm(u)>epsilon
     delta_u=u_hat-u;
     c_hat=c_hat-delta_u(ap);       %Update estimate of c
     u_hat=c_hat*Asn;               %Update estimate of u
end;
```

REFERENCES

[1] WO 00/58914

[2] Using MATLAB, MathWorks Inc, 1996, pp. 4-2 – 4-3, 4-13 – 4-14

CLAIMS

1. An artificial vision system training method, including the steps:

forming a feature matrix including feature sample vectors and a corresponding response sample vector including response sample scalars;

determining a linkage vector linking said feature matrix to said response sample vector, **characterized** by an iterative linkage vector determining method including the steps:

determining a response sample vector error estimate in the response sample vector domain;

transforming said response sample vector error estimate into a corresponding linkage vector error estimate in the linkage vector domain;

determining a linkage vector estimate in the linkage vector domain by using said linkage vector error estimate;

transforming said linkage vector estimate into a corresponding response sample vector estimate in the response sample vector domain;

repeating the previous steps until the response sample vector error estimate is sufficiently small.

2. The method of claim 1, **characterized** by an iteration step including:

determining a response sample vector error estimate representing the difference between a current response sample vector estimate and said response sample vector;

using the transpose of said feature matrix for transforming said response sample vector error estimate into said linkage vector error estimate;

forming an updated linkage vector estimate by subtracting said linkage vector error estimate from a current linkage vector estimate;

using said feature matrix for transforming said updated linkage vector estimate into an updated response sample vector estimate.

3. The method of claim 1, **characterized** by an iteration step including:

using said feature matrix for transforming a current linkage vector estimate into an updated response sample vector estimate;

determining a response sample vector error estimate representing the difference between said updated response sample vector estimate and said response sample vector;

using the transpose of said feature matrix for transforming said response sample vector error estimate into a linkage vector error estimate;

forming an updated linkage vector estimate by subtracting said linkage vector error estimate from said current linkage vector estimate.

4. The method of claim 2 or 3, **characterized** by feature normalizing said linkage vector error estimate.

5. The method of claim 4, **characterized** by a feature normalization represented by the diagonal elements of the matrix:

$$\mathbf{N}^F = \begin{pmatrix} 1/\sum_{n=1}^{N}\left(\sum_{h=1}^{H} a_h^n\right)a_1^n & 0 & \cdots & 0 \\ 0 & 1/\sum_{n=1}^{N}\left(\sum_{h=1}^{H} a_h^n\right)a_2^n & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & 1/\sum_{n=1}^{N}\left(\sum_{h=1}^{H} a_h^n\right)a_H^n \end{pmatrix}$$

where

a_h^n denote the elements of said feature matrix,

N represents the number of feature sample vectors, and

H represents the number of components in each feature sample vector.

6. The method of claim 2 or 3, **characterized** by sample normalizing said updated response sample vector estimate.

7. The method of claim 6, **characterized** by a sample normalization represented by the diagonal elements of the matrix:

$$\mathbf{N}^S = \begin{pmatrix} 1/\sum_{h=1}^{H}\left(\sum_{n=1}^{N} a_h^n\right)a_h^1 & 0 & \cdots & 0 \\ 0 & 1/\sum_{h=1}^{H}\left(\sum_{n=1}^{N} a_h^n\right)a_h^2 & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & 1/\sum_{h=1}^{H}\left(\sum_{n=1}^{N} a_h^n\right)a_h^N \end{pmatrix}$$

where

a_h^n denote the elements of said feature matrix,

N represents the number of feature sample vectors, and

H represents the number of components in each feature sample vector.

8. The method of claim 2 or 3, **characterized** by

feature normalizing said linkage vector error estimate; and

sample normalizing said updated response sample vector estimate.

9. The method of claim 8, **characterized** by a feature normalization represented by the diagonal elements of the matrix:

$$\mathbf{N}^F = \begin{pmatrix} 1/\sum_{n=1}^{N} a_1^n & 0 & \cdots & 0 \\ 0 & 1/\sum_{n=1}^{N} a_2^n & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & 1/\sum_{n=1}^{N} a_H^n \end{pmatrix}$$

and a sample normalization represented by the diagonal elements of the matrix:

$$\mathbf{N}^S = \begin{pmatrix} 1/\sum_{h=1}^{H} a_h^1 & 0 & \cdots & 0 \\ 0 & 1/\sum_{h=1}^{H} a_h^2 & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & 1/\sum_{h=1}^{H} a_h^N \end{pmatrix} \quad ($$

where

a_h^n denote the elements of said feature matrix,

N represents the number of feature sample vectors, and

H represents the number of components in each feature sample vector.

10. The method of any of the preceding claims 2-9, **characterized** by selectively replacing said feature matrix by an approximate feature matrix, in which only the maximum element is retained in each row and all other elements are replaced by zero.

11. The method of any of the preceding claims 2-10, **characterized** by selectively replacing said feature matrix by an approximate feature matrix, in which only the maximum element is retained in each column and all other elements are replaced by zero.

12. The method of any of the preceding claims, **characterized** in that

all non-zero elements of said feature matrix have the same sign; and

all non-zero elements of said response sample vector have the same sign.

13. An artificial vision system training method, including the steps:

forming a feature matrix including feature sample vectors and a corresponding response sample matrix including response sample vectors;

determining a linkage matrix linking said feature matrix to said response sample matrix, **characterized** by an iterative linkage matrix determining method including the steps:

determining a response sample matrix error estimate in the response sample matrix domain;

transforming said response sample matrix error estimate into a corresponding linkage matrix error estimate in the linkage matrix domain;

determining a linkage matrix estimate in the linkage matrix domain by using said linkage matrix error estimate;

transforming said linkage matrix estimate into a corresponding response sample matrix estimate in the response sample matrix domain;

repeating the previous steps until the response sample matrix error estimate is sufficiently small.

14. An associative network training method, including the steps:

forming a feature matrix including feature sample vectors and a corresponding response sample vector including response sample scalars;

determining a linkage vector linking said feature matrix to said response sample vector, **characterized** by an iterative linkage vector determining method including the steps:

determining a response sample vector error estimate in the response sample vector domain;

transforming said response sample vector error estimate into a corresponding linkage vector error estimate in the linkage vector domain;

determining a linkage vector estimate in the linkage vector domain by using said linkage vector error estimate;

transforming said linkage vector estimate into a corresponding response sample vector estimate in the response sample vector domain;

repeating the previous steps until the response sample vector error estimate is sufficiently small.

15. An associative network training method, including the steps:

forming a feature matrix including feature sample vectors and a corresponding response sample matrix including response sample vectors;

determining a linkage matrix linking said feature matrix to said response sample matrix, **characterized** by an iterative linkage matrix determining method including the steps:

determining a response sample matrix error estimate in the response sample matrix domain;

transforming said response sample matrix error estimate into a corresponding linkage matrix error estimate in the linkage matrix domain;

determining a linkage matrix estimate in the linkage matrix domain by using said linkage matrix error estimate;

transforming said linkage matrix estimate into a corresponding response sample matrix estimate in the response sample matrix domain;

repeating the previous steps until the response sample matrix error estimate is sufficiently small.

16. An artificial vision system linkage vector training apparatus, including means for forming a feature matrix including feature sample vectors and a corresponding response sample vector including response sample scalars, **characterized** by:

means (10) for determining a response sample vector error estimate in the response sample vector domain;

means (12) for transforming said response sample vector error estimate into a corresponding linkage vector error estimate in the linkage vector domain;

means (14, 16) for determining a linkage vector estimate in the linkage vector domain by using said linkage vector error estimate; and

means (18) for transforming said linkage vector estimate into a corresponding response sample vector estimate in the response sample vector domain.

17. An artificial vision system linkage matrix training apparatus, including means for forming a feature matrix including feature sample vectors and a corresponding response sample matrix including response sample vectors, **characterized** by:

means (10) for determining a response sample matrix error estimate in the response sample matrix domain;

means (12) for transforming said response sample matrix error estimate into a corresponding linkage matrix error estimate in the linkage matrix domain;

means (14, 16) for determining a linkage matrix estimate in the linkage matrix domain by using said linkage matrix error estimate; and

means (18) for transforming said linkage matrix estimate into a corresponding response sample matrix estimate in the response sample matrix domain.

18. An associative network linkage vector training apparatus, including means for forming a feature matrix including feature sample vectors and a corresponding response sample vector including response sample scalars, **characterized** by:

means (10) for determining a response sample vector error estimate in the response sample vector domain;

means (12) for transforming said response sample vector error estimate into a corresponding linkage vector error estimate in the linkage vector domain;

means (14, 16) for determining a linkage vector estimate in the linkage vector domain by using said linkage vector error estimate; and

means (18) for transforming said linkage vector estimate into a corresponding response sample vector estimate in the response sample vector domain.

19. An associative network system linkage matrix training apparatus, including means for forming a feature matrix including feature sample vectors and a

corresponding response sample matrix including response sample vectors, **characterized** by:

means (10) for determining a response sample matrix error estimate in the response sample matrix domain;

means (12) for transforming said response sample matrix error estimate into a corresponding linkage matrix error estimate in the linkage matrix domain;

means (14, 16) for determining a linkage matrix estimate in the linkage matrix domain by using said linkage matrix error estimate; and

means (18) for transforming said linkage matrix estimate into a corresponding response sample matrix estimate in the response sample matrix domain.

20. A computer program product for determining a linkage vector linking a feature matrix to a response sample vector, comprising program elements for performing the steps:

determining a response sample vector error estimate in the response sample vector domain;

transforming said response sample vector error estimate into a corresponding linkage vector error estimate in the linkage vector domain;

determining a linkage vector estimate in the linkage vector domain by using said linkage vector error estimate;

transforming said linkage vector estimate into a corresponding response sample vector estimate in the response sample vector domain;

repeating the previous steps until the response sample vector error estimate is sufficiently small.

21. A computer program product for determining a linkage matrix linking a feature matrix to a response sample matrix, comprising program elements for performing the steps:

determining a response sample matrix error estimate in the response sample matrix domain;

transforming said response sample matrix error estimate into a corresponding linkage matrix error estimate in the linkage matrix domain;

determining a linkage matrix estimate in the linkage matrix domain by using said linkage matrix error estimate;

transforming said linkage matrix estimate into a corresponding response sample matrix estimate in the response sample matrix domain;

repeating the previous steps until the response sample matrix error estimate is sufficiently small.

1/2

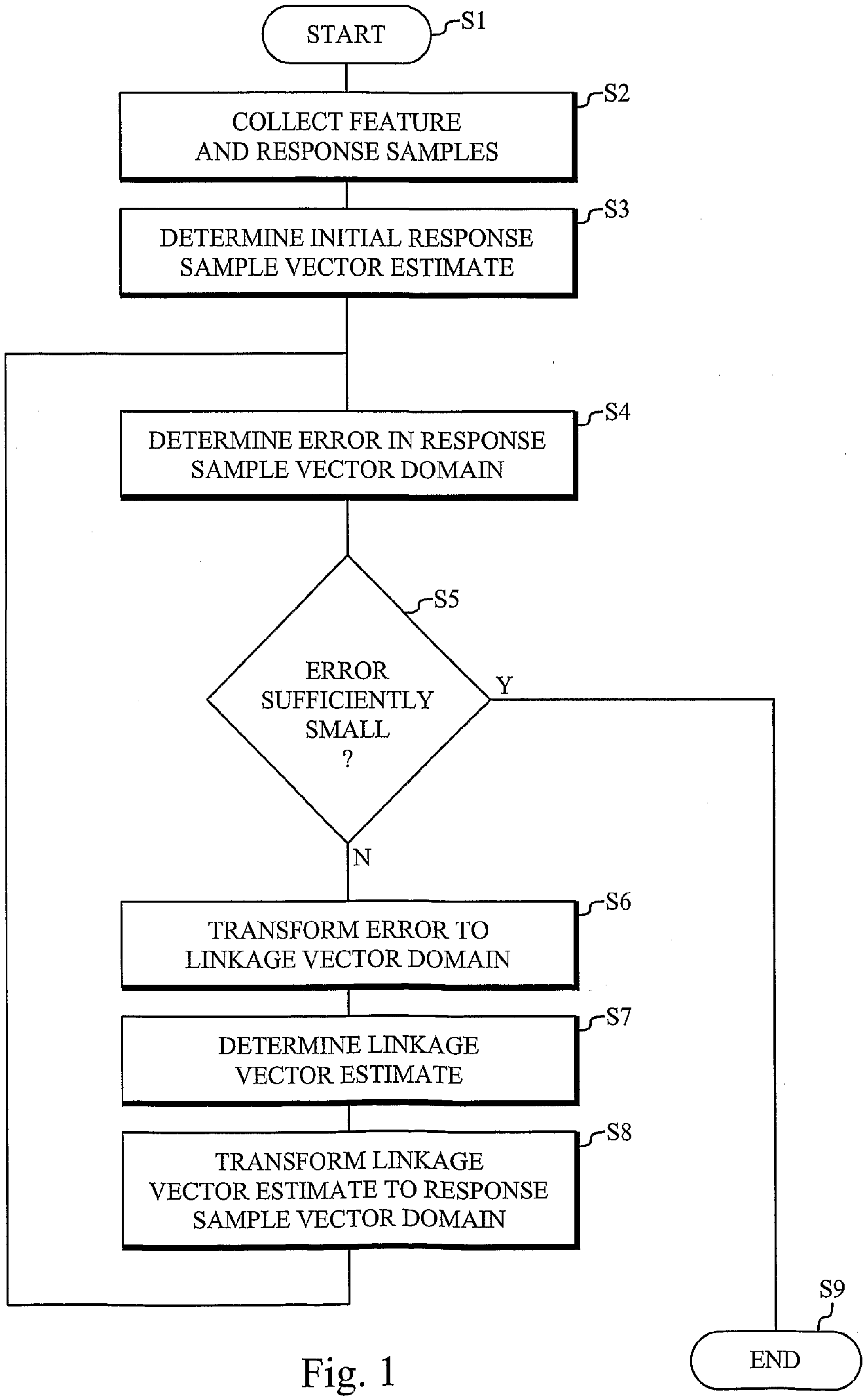

START
S1
COLLECT FEATURE AND RESPONSE SAMPLES
S2
DETERMINE INITIAL RESPONSE SAMPLE VECTOR ESTIMATE
S3
DETERMINE ERROR IN RESPONSE SAMPLE VECTOR DOMAIN
S4
ERROR SUFFICIENTLY SMALL ?
S5
Y
N
TRANSFORM ERROR TO LINKAGE VECTOR DOMAIN
S6
DETERMINE LINKAGE VECTOR ESTIMATE
S7
TRANSFORM LINKAGE VECTOR ESTIMATE TO RESPONSE SAMPLE VECTOR DOMAIN
S8
S9
END

Fig. 1

2/2

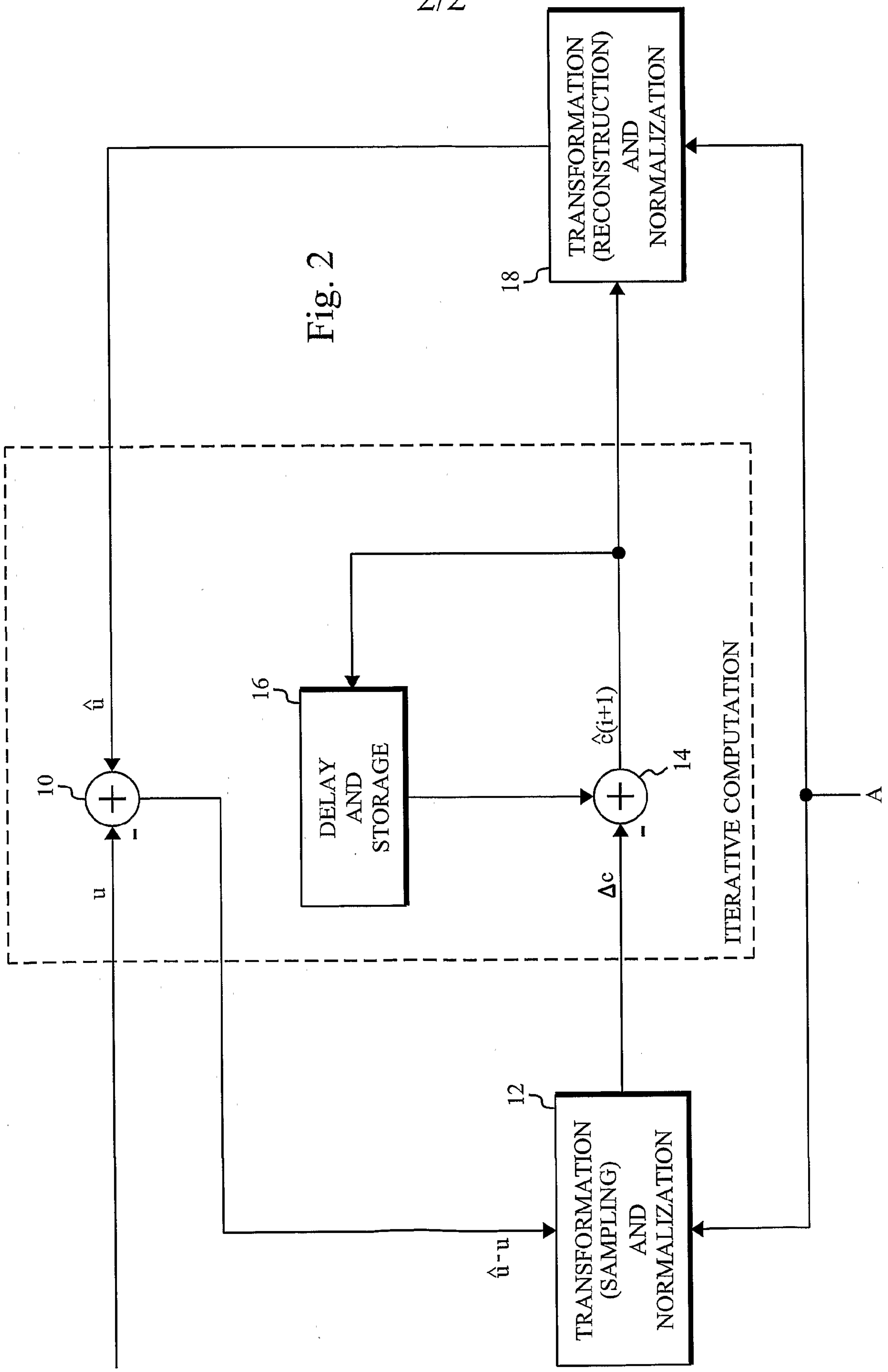

TRANSFORMATION (RECONSTRUCTION) AND NORMALIZATION
18
DELAY AND STORAGE
16
10
14
ITERATIVE COMPUTATION
TRANSFORMATION (SAMPLING) AND NORMALIZATION
12
û
u
ĉ(i+1)
Δc
û-u
A

Fig. 2

INTERNATIONAL SEARCH REPORT

International application No.
PCT/SE 01/02285

A. CLASSIFICATION OF SUBJECT MATTER

IPC7: G06F 15/18, G06T 1/00, G06K 9/66
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC7: G06F, G06K, G06T

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

SE,DK,FI,NO classes as above

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

WPI DATA, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 6134537 A (Y-H PAO ET AL), 17 October 2000 (17.10.00), figure 1, abstract --	1-21
A	US 5249259 A (R.L. HARVEY), 28 Sept 1993 (28.09.93), figure 8, abstract --	1-21
A	WO 0055790 A2 (MARKETSWITCH CORP), 21 Sept 2000 (21.09.00), figure 1, abstract --	1-21
A	WO 0058914 A1 (GRANLUND, GÖSTA), 5 October 2000 (05.10.00) -- --------	1-21

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance: the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance: the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
15 February 2002	20 -02- 2002
Name and mailing address of the ISA/ Swedish Patent Office Box 5055, S-102 42 STOCKHOLM Facsimile No. +46 8 666 02 86	Authorized officer Patrik Blidefalk/AE Telephone No. +46 8 782 25 00

INTERNATIONAL SEARCH REPORT

International application No.
PC/SE01/02285

Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☒ Claims Nos.: 1-21
because they relate to subject matter not required to be searched by this Authority, namely:

mathematical theories.
See PCT-rule 39.1(i).
However, a search for prior art is done, but any opininion of the documents is not formed.

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest.

☐ No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT
Information on patent family members

28/01/02

International application No.
PCT/SE 01/02285

Patent document cited in search report			Publication date	Patent family member(s)			Publication date
US	6134537	A	17/10/00	AU	737276	B	16/08/01
				AU	1912699	A	05/07/99
				BR	9813585	A	26/12/01
				CN	1282435	T	31/01/01
				EP	1038261	A	27/09/00
				US	6212509	B	03/04/01
				US	2001032198	A	18/10/01
				WO	9931624	A	24/06/99
				US	5734796	A	31/03/98
US	5249259	A	28/09/93	WO	9111771	A	08/08/91
WO	0055790	A2	21/09/00	AU	3884000	A	04/10/00
WO	0058914	A1	05/10/00	AU	4938699	A	16/10/00
				AU	5073699	A	05/01/00
				BR	9911333	A	03/04/01
				EP	1095332	A	02/05/01
				EP	1171846	A	16/01/02
				SE	513728	C	30/10/00
				SE	9901110	A	27/09/00