

(12) DEMANDE INTERNATIONALE PUBLIÉE EN VERTU DU TRAITÉ DE COOPÉRATION  
EN MATIÈRE DE BREVETS (PCT)

(19) Organisation Mondiale de la Propriété  
Intellectuelle  
Bureau international



(43) Date de la publication internationale  
27 septembre 2007 (27.09.2007)

PCT

(10) Numéro de publication internationale  
**WO 2007/107659 A2**

- (51) Classification internationale des brevets : **Non classée**
- (21) Numéro de la demande internationale :  
PCT/FR2007/050908
- (22) Date de dépôt international : 9 mars 2007 (09.03.2007)
- (25) Langue de dépôt : français
- (26) Langue de publication : français
- (30) Données relatives à la priorité :  
0602452 21 mars 2006 (21.03.2006) FR
- (71) Déposant (pour tous les États désignés sauf US) :  
**FRANCE TELECOM** [FR/FR]; 6 Place d'Alleray,  
F-75015 Paris (FR).
- (72) Inventeurs; et
- (75) Inventeurs/Déposants (pour US seulement) : **RAGOT,  
Stéphane** [FR/FR]; 23 Bd Thalassa, Appt 5, Résidence  
de la Plage, F-22700 Perros-Guirec (FR). **GUILLAUME,**

Cyril [FR/FR]; La Haute Jumeaubretière, F-85500 Beau-  
repaire (FR).

(74) Mandataire : **FROGER, Marie-Hélène**; France Télécom  
R & D/PIV/BREVETS, 38-40 Rue du Général Leclerc,  
F-92794 Issy Les Moulineaux Cedex 9 (FR).

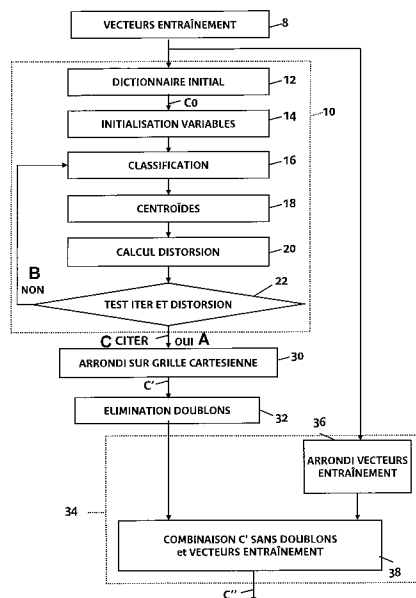
(81) États désignés (sauf indication contraire, pour tout titre de  
protection nationale disponible) : AE, AG, AL, AM, AT,  
AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN,  
CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI,  
GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS,  
JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS,  
LT, LU, LY, MA, MD, MG, MK, MN, MW, MX, MY, MZ,  
NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU,  
SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR,  
TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) États désignés (sauf indication contraire, pour tout titre  
de protection régionale disponible) : ARIPO (BW, GH,

[Suite sur la page suivante]

(54) Title: RESTRAINED VECTOR QUANTISATION

(54) Titre : QUANTIFICATION VECTORIELLE CONTRAINTE



(57) Abstract: The invention relates to a method for generating a dictionary for the vector quantisation of a signal. Said method comprises a step (10) of the statistical analysis of driving vectors representing the signal determining a finished set of code vectors (CITER) representing said driving vectors. The inventive method is characterised in that it also comprises a step (3) of modifying the finished set of code vectors in order to impose a minimum distance between the code vectors modified two-by-two, the set of said modified code vectors forming the dictionary.

(57) Abrégé : Ce procédé de génération d'un dictionnaire de quantification vectorielle d'un signal est du type comportant une étape (10) d'analyse statistique de vecteurs d'entraînement représentatifs du signal déterminant un ensemble fini de vecteurs de code (CITER) représentant lesdits vecteurs d'entraînement. Il est caractérisé en ce qu'il comporte en outre une étape (30) de modification dudit ensemble fini de vecteurs de code pour imposer une distance minimale entre les vecteurs de code modifiés deux à deux, l'ensemble de ces vecteurs de code modifiés formant ledit dictionnaire.

- A yes  
B no  
C CITER  
8 driving vectors  
12 initial dictionary  
14 initialisation of variables  
16 classification  
18 centroids  
20 calculation of distortion  
22 test ITER and distortion  
30 rounding on cartesian grid  
32 elimination of doubles  
36 rounding driving vectors  
38 combination C' without doubles and driving vectors

WO 2007/107659 A2



GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), eurasien (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), européen (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

**Déclaration en vertu de la règle 4.17 :**

— *relative à la qualité d'inventeur (règle 4.17.iv)*

**Publiée :**

— *sans rapport de recherche internationale, sera republiée dès réception de ce rapport*

*En ce qui concerne les codes à deux lettres et autres abréviations, se référer aux "Notes explicatives relatives aux codes et abréviations" figurant au début de chaque numéro ordinaire de la Gazette du PCT.*

## QUANTIFICATION VECTORIELLE CONTRAINTE

La présente invention concerne les dictionnaires de quantification pour le codage et le décodage de signaux tels que les signaux audio, vidéo, et plus généralement les signaux multimédia, en vue de leur stockage ou de leur  
5 transmission.

Plus précisément, la présente invention propose une solution au problème de la quantification vectorielle statistique non-structurée intégrant la connaissance a priori d'un seuil de distorsion.

Une solution très répandue pour le traitement des signaux notamment  
10 numériques est la quantification vectorielle (QV) qui, de manière générale, représente un vecteur d'entrée par un vecteur de même dimension choisi dans un ensemble fini. Cet ensemble fini est appelé alphabet de reproduction ou dictionnaire ou encore répertoire, et ses éléments des vecteurs de code, des mots de code, des points de sortie ou des représentants.

En quantification vectorielle, un bloc de  $n$  échantillons d'un signal est  
15 traité comme un vecteur de dimension  $n$ . Le vecteur est codé en choisissant, dans un dictionnaire fini, le vecteur de code qui lui « ressemble » le plus. Par exemple, une recherche exhaustive est faite parmi tous les éléments du dictionnaire pour sélectionner l'élément du dictionnaire qui minimise une  
20 mesure de distance entre cet élément et le vecteur d'entrée.

En général, les dictionnaires de quantification vectorielle sont conçus à partir de l'analyse d'échantillons du signal à coder formant une séquence d'entraînement, à l'aide de méthodes statistiques telles que les algorithmes de Lloyd-Max généralisé (GLA: Generalized Lloyd-Max Algorithm) ou LBG (pour  
25 Linde, Buzo et Gray). Un tel type d'algorithme est décrit par exemple dans l'ouvrage Gersho A., Gray R.M, « Vector Quantization and Signal Compression », Kluwer Academic Publishers, 1992.

A partir de la séquence d'entraînement, et d'un dictionnaire initial, le dictionnaire est construit de façon itérative. Chaque itération comprend une  
30 étape de classification des vecteurs d'entraînement pour construire des régions de quantification de la séquence d'entraînement selon la règle du plus proche

voisin, et une étape d'amélioration du dictionnaire en remplaçant les anciens vecteurs de code par les centroïdes des régions, selon la règle dite des centroïdes.

Pour éviter la convergence vers un minimum local de cet algorithme itératif déterministe, des variantes dites de relaxation stochastique (SKA: Stochastic K-means algorithm) ont été proposées en introduisant une part d'aléatoire dans l'étape de construction des centroïdes et/ou dans celle de construction des classes. Une telle modification est décrite par exemple dans la publication Kovesi, B.; Saoudi, S.; Boucher, J.M.; Reguly, Z., « A fast robust stochastic algorithm for vector quantizer design for nonstationary channels », 5  
10 IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 1, pp. 269-272, 1995.

Les dictionnaires, ou quantificateurs vectoriels statistiques, ainsi obtenus ne possèdent aucune structure particulière, ce qui rend leur 15 exploration coûteuse en calculs et gourmande en mémoire. De fait, l'utilisation, notamment en temps réel, de tels dictionnaires obtenus par quantification vectorielle statistique classique est souvent limitée à des codages de faible dimension et/ou des bas débits de l'ordre de 4 à 10 bits par vecteur.

De manière générale, la quantification vectorielle statistique exploite la 20 forme de la distribution de probabilité du signal à traiter et cela se retrouve dans la distribution des vecteurs de code du dictionnaire d'un quantificateur vectoriel statistique. Les algorithmes de construction utilisés dans la quantification vectorielle visent la minimisation de la distorsion moyenne; ils ont donc tendance à placer beaucoup de vecteurs de code dans les zones à forte 25 densité. Cette concentration des centroïdes dans les zones de forte densité du signal à coder n'est pas toujours pertinente d'un point de vue perceptuel. Ainsi, il se peut que deux vecteurs de code d'une zone de forte densité soient si proches que la distorsion générée par la substitution de l'un par l'autre ne soit pas perceptible.

30 De plus, pour des quantificateurs à faible résolution, c'est-à-dire avec peu de vecteurs de code, cette concentration se fait au détriment de zones

moins denses de la distribution de probabilité du signal à coder qui sont mal représentées, c'est-à-dire mal codées.

Pour s'affranchir des contraintes de taille et de dimension, plusieurs variantes de quantification vectorielle statistiques furent développées, notamment pour tenter de remédier à l'absence de structure du dictionnaire et ainsi réduire la complexité au détriment de la qualité. Cependant, le compromis performance-complexité est amélioré, ce qui permet d'accroître la plage des résolutions et/ou des dimensions sur laquelle la quantification vectorielle peut être appliquée efficacement. Beaucoup de schémas de quantificateurs vectoriels contraints ont été proposés dans la littérature. On peut notamment citer le quantificateur vectoriel en arbre, le quantificateur vectoriel à étages multiples, le quantificateur vectoriel « produit cartésien » ou encore le quantificateur vectoriel « gain/orientation », cas particulier du quantificateur vectoriel produit cartésien.

Comme en quantification vectorielle non-structurée, les dictionnaires obtenus par des méthodes contraintes présentent toujours une quantité importante de vecteurs de code proches les uns des autres et notamment de vecteurs de code dont les différences sont non discernables par le récepteur. Ceci conduit à une qualité non optimale.

Une autre approche, appelée quantification vectorielle algébrique, a aussi été proposée. La quantification vectorielle algébrique utilise des dictionnaires fortement structurés, issus de réseaux réguliers de points ou des codes correcteurs d'erreur. Un exemple simple de réseau régulier est le réseau cubique, formé par exemple d'un ensemble de points à coordonnées entières.

Grâce aux propriétés algébriques de leurs dictionnaires, les quantificateurs vectoriels algébriques sont simples à mettre en œuvre et n'ont pas à être stockés en mémoire. L'exploitation de la structure régulière de ces dictionnaires permet le développement d'algorithmes de recherche optimaux et rapides et de mécanismes pour associer l'index à un vecteur de code correspondant, par exemple par une formule.

Les techniques de quantification vectorielle algébrique développées dans l'état de l'art aboutissent à des dictionnaires qui sont des sous-ensembles de réseaux réguliers.

Un des paramètres principaux d'un réseau régulier est sa distance  
5 quadratique minimale entre deux points sur lesquels sont placés les vecteurs de code. Pour faciliter l'indexation, le dictionnaire est généralement choisi comme l'intersection du réseau avec un polytope régulier, tel qu'un hypercube ou une hypersphère, ou avec sa surface. Les points en dehors de cette intersection sont alors des outliers. Pour un débit donné, c'est-à-dire un  
10 nombre fixé de points de l'intersection, limiter le nombre d'outliers conduit à augmenter la distance minimale entre deux vecteurs de code.

Si les quantificateurs vectoriels algébriques sont moins complexes à mettre en oeuvre et nécessitent moins de mémoire, ils ne sont optimaux que pour coder des signaux avec une distribution uniforme.

15 Il faut aussi noter que les opérations d'indexation des vecteurs de code, ou numérotage, et les opérations inverses de décodage nécessitent plus de calculs que dans le cas des quantificateurs vectoriels statistiques, pour lesquels ces opérations sont effectuées par de simples lectures de tables.

Enfin, une autre forme de quantification vectorielle combine la  
20 quantification algébrique et la quantification statistique. C'est le cas du quantificateur vectoriel fin-grossier « Fine-Coarse Vector Quantization » décrit dans la publication « Moayeri, N., Neuhoff D.L. ; Stark, W.E, « Fine-coarse vector quantization » IEEE Trans. on Information Theory, Vol. 37, Issue 4, pp. 1503-1515, July 1991 ». Cette quantification comprend deux quantificateurs. Le  
25 premier quantificateur à haute résolution, ou à résolution fine, est généralement bien structuré et peu complexe à mettre en oeuvre tandis que le deuxième, à résolution grossière est un quantificateur vectoriel statistique non structuré. Le premier étage sert de pré-calcul pour accélérer la recherche dans le second étage.

30 Le quantificateur « Fine-coarse » souffre des mêmes inconvénients que ceux indiqués précédemment pour la quantification vectorielle statistique. En effet, le lien entre le quantificateur à résolution fine et le quantificateur à

résolution grossière est une simple table de correspondance et donc, même si le quantificateur vectoriel à haute résolution est un sous-ensemble fini d'un réseau régulier avec une distance minimale entre les vecteurs de code, cette contrainte n'est pas imposée au quantificateur vectoriel à résolution grossière  
5 qui est construit par les techniques classiques de quantification vectorielle statistique.

Il apparaît donc que la quantification vectorielle statistique aboutit à des dictionnaires dont la répartition des points est mal adaptée, une partie de ces points étant redondants car indiscernables au niveau du récepteur.

10 Inversement, la quantification vectorielle algébrique aboutit à des dictionnaires dont la distribution n'est adaptée que pour des signaux à distribution de probabilité uniforme et aboutit à dégrader la qualité du codage pour les autres signaux.

L'invention a notamment pour avantage de définir un procédé  
15 d'obtention d'un dictionnaire par quantification vectorielle statistique adapté au signal à coder et prenant en compte les seuils perceptuels.

A cet effet, l'invention a pour objet un procédé de génération d'un dictionnaire de quantification vectorielle d'un signal, du type comportant une étape d'analyse statistique de vecteurs d'entraînement représentatifs du signal  
20 déterminant un ensemble fini de vecteurs de code représentants lesdits vecteurs d'entraînement, caractérisé en ce que le procédé comporte en outre une étape de modification dudit ensemble fini de vecteurs de code pour imposer une distance minimale entre les vecteurs de code modifiés deux à deux, ces vecteurs modifiés formant ledit dictionnaire.

25 Ainsi, l'invention aboutit à un dictionnaire adapté au signal du fait d'une étape d'analyse statistique des vecteurs d'entraînement mais avec une distance minimale entre les vecteurs de code, ce qui permet de prendre en compte le fait que des vecteurs de code trop proches sont indiscernables au niveau de la perception.

30 Selon d'autres caractéristiques de l'invention, l'étape d'analyse statistique comprend une sous-étape de création d'un dictionnaire initial, une sous-étape de classification des vecteurs d'entraînement à partir du

dictionnaire initial pour former des régions de quantification et une sous-étape de détermination d'un centroïde pour chaque région, lesdits centroïdes étant lesdits vecteurs de code. Un tel mode de réalisation correspond à une mise en œuvre particulière d'analyse statistique.

5           Avantageusement, ladite sous-étape de création d'un dictionnaire initial est réalisée à partir desdits vecteurs d'entraînement et comprend le remplacement de chaque vecteur d'entraînement par un vecteur arrondi choisi sur un réseau régulier de points et l'élimination sélective de vecteurs arrondis en fonction de leur fréquence d'apparition.

10           Dans un mode particulier de réalisation de l'invention, l'étape d'analyse statistique comporte en outre une sous-étape de calcul de la distorsion du dictionnaire initial et une sous-étape de comparaison de cette distorsion avec un seuil de tolérance, ladite étape d'analyse statistique étant répétée si ladite comparaison est négative. Une telle approche itérative permet d'obtenir un  
15 dictionnaire initial de meilleure qualité.

          Dans une variante, l'étape de modification comprend le remplacement de chacun desdits vecteurs de code par un vecteur de code voisin choisi sur un réseau régulier de points. Un tel mode de réalisation permet d'assurer une distance minimale entre les vecteurs de code.

20           Avantageusement, le procédé comporte en outre une étape d'élimination des doublons parmi lesdits vecteurs de code modifiés ce qui a pour conséquence de réduire la taille du dictionnaire.

          Dans un mode de réalisation particulier, ladite étape d'élimination des doublons comporte une sous-étape de recherche des doublons par  
25 comparaison des vecteurs de code modifiés entre eux et une sous-étape de suppression des doublons.

          Avantageusement, le procédé comporte une étape de fusion des vecteurs de code modifiés avec les vecteurs d'entraînement présentant au moins ladite distance minimale entre eux et avec lesdits vecteurs de code  
30 modifiés afin d'améliorer la répartition des vecteurs de code du dictionnaire.

          Dans un mode de réalisation particulier, cette étape de fusion comporte une sous-étape de remplacement de chaque vecteur d'entraînement par un

vecteur arrondi choisi sur un réseau régulier de points et une sous-étape de combinaison des vecteurs de code modifiés et des vecteurs d'entraînement arrondis. Ainsi, les vecteurs d'entraînement arrondi qui sont combinés aux vecteurs de code modifiés, présentent également une distance minimale deux  
5 à deux.

Dans encore une variante, ladite étape de fusion comprend en outre l'ordonnement des vecteurs d'entraînement arrondis selon leur fréquence d'apparition et la sélection d'un nombre fini de ces vecteurs pour les combiner avec les vecteurs de code modifiés.

10 Dans un mode de réalisation, ladite sous-étape de combinaison comprend l'adjonction auxdits vecteurs de code modifiés d'un nombre fini de vecteurs d'entraînement arrondis distincts desdits vecteurs de code modifiés.

Dans une variante, le réseau régulier de points a un pas correspondant à un niveau de sensibilité d'un récepteur destinataire dudit signal, ce qui  
15 permet d'imposer une distance minimale entre deux vecteurs de code correspondant à une réalité perceptuelle.

Dans encore un autre mode de réalisation, le procédé comporte en outre une phase d'optimisation dudit dictionnaire.

Par exemple, la phase d'optimisation comporte une étape de calcul de  
20 la distorsion du dictionnaire et une étape de comparaison de cette distorsion avec un seuil de tolérance, lesdites étapes d'analyse statistique et de modification étant répétées si ladite comparaison est négative.

Alternativement, la phase d'optimisation comporte au moins une  
25 itération du procédé mis en œuvre à partir du dictionnaire formé desdits vecteurs de code modifiés.

Par ailleurs, l'invention a également pour objet un programme d'ordinateur comportant des instructions de code pour mettre en œuvre le procédé tel que mentionné précédemment, ainsi qu'un dispositif comportant des moyens tels que, par exemple, une mémoire travail et un processeur, pour  
30 exécuter ce programme d'ordinateur et générer ainsi un dictionnaire de quantification vectorielle au sens de l'invention.

La présente invention vise aussi des procédés de codage et de décodage, ainsi qu'un décodeur utilisant un dictionnaire obtenu selon l'invention.

L'invention sera mieux comprise à la lumière de la description qui suit, faite à titre d'exemple non limitatif en référence aux figures annexées sur lesquelles :

- la figure 1 est un organigramme du procédé de l'invention ;
- les figures 2 et 3 sont des représentations d'un dictionnaire à différentes étapes du procédé de l'invention ;
- 10 - la figure 4 est un organigramme d'un procédé d'optimisation ;
- les figures 5 et 6 sont des schémas bloc d'un codeur et d'un décodeur mettant en œuvre le dictionnaire obtenu grâce au procédé de l'invention.

En référence à la figure 1, on va maintenant décrire un procédé d'obtention d'un dictionnaire de quantification vectorielle selon l'invention.

Dans le mode de réalisation décrit, ce procédé est mis en œuvre sur des vecteurs d'entraînement représentatifs d'un signal audiophonique numérisé tel qu'un signal de parole.

Ce procédé comprend tout d'abord une étape 10 d'analyse statistique des vecteurs d'entraînement. Dans le mode de réalisation décrit, l'étape 10 correspond à l'application de l'algorithme de Lloyd-Max généralisé.

L'étape 10 débute par une sous-étape 12 de création d'un dictionnaire initial noté C0. Ce dictionnaire est créé à partir de l'analyse des vecteurs d'entraînement ou par d'autres moyens connus en soi tels qu'une sélection aléatoire d'un nombre fini des vecteurs d'entraînement, ou des algorithmes dits « de splitting », « algorithme LBG », ou autre. Ce dictionnaire initial C0 est d'une taille déterminée.

Dans le mode de réalisation décrit, le procédé comprend ensuite une sous-étape 14 d'initialisation de variables courantes et notamment d'initialisation d'une variable ITER correspondant à un nombre d'itération et d'une variable DITER correspondant à une valeur de distorsion.

L'étape 10 comporte ensuite une sous-étape 16 de classification des vecteurs d'entraînement par rapport au dictionnaire initial C0 pour former des régions de classification et une sous-étape 18 de détermination d'un centroïde pour chaque région.

5 Plus précisément, lors de la sous-étape 16 de classification, pour chaque vecteur d'entraînement, on parcourt l'ensemble du dictionnaire courant, c'est-à-dire du dictionnaire C0 modifié par itérations successives, et l'on choisit le vecteur de code du dictionnaire minimisant l'erreur quadratique avec le vecteur d'entraînement.

10 La sous-étape 18 est réalisée de la manière suivante : pour chaque classe ou région  $V_i$ , on calcule les nouveaux centroïdes.

On définit  $V_i$  l'ensemble des vecteurs d'entraînement  $x_j$  vérifiant  $Q(x_j) = y_i$ , où  $Q()$  est la fonction de quantification et  $y_i$  est un vecteur de code du dictionnaire.

15 Soit  $N_i$  le nombre de vecteurs de  $V_i$ , et  $V_i(j,k)$  le  $k$ ème élément du  $j$ ème vecteur de  $V_i$ .

On a, pour le  $k$ ème élément du centroïde  $y_i$  :

$$y_i[k] = \frac{1}{N_i} \sum_{j=0}^{N_i} V_i[j,k]$$

20 Lors d'une sous-étape 20, une valeur de distorsion pour le dictionnaire courant est calculée.

Lors de cette sous-étape 20, les vecteurs d'entraînement sont quantifiés avec le dictionnaire courant fourni par les étapes précédentes. L'erreur quadratique entre les vecteurs d'entraînement et leur version quantifiée constitue la mesure de distorsion.

25 Les sous-étapes 16, 18 et 20 sont ensuite répétées de manière itérative jusqu'à ce qu'un nombre maximal d'itérations ITERMAX soit atteint ou que la distorsion soit inférieure à un seuil de tolérance, les tests étant effectués lors d'une sous-étape 22.

30 Comme indiqué précédemment, dans l'exemple, cette étape 10, avec notamment la répétition des sous-étapes 16, 18, 20 et 22, correspond à l'application de l'algorithme de Lloyd-Max généralisé.

A l'issue de l'étape 10, le procédé délivre ainsi un dictionnaire CITER formé par une analyse statistique itérative des vecteurs d'entraînement.

Une représentation en deux dimensions du dictionnaire CITER est donnée sur la figure 2 sur laquelle les vecteurs de code sont représentés par un nuage de points. Sur cette représentation, les points gris correspondent aux vecteurs d'entraînement ou vecteurs du signal et les points noirs aux vecteurs du dictionnaire. On voit que la répartition des vecteurs du dictionnaire CITER suit la densité des vecteurs d'entraînement et que de nombreux vecteurs sont proches les uns des autres.

Le procédé comporte ensuite une étape 30 de modification du dictionnaire CITER pour imposer une distance minimale entre les vecteurs de code deux à deux.

Cette étape 30 de modification comprend le remplacement de chacun des vecteurs de code par un vecteur voisin choisi sur un réseau régulier de points tel qu'une grille cartésienne. Ces vecteurs modifiés sont également appelés vecteurs arrondis. Dans le mode de réalisation décrit, le vecteur voisin choisi est le vecteur placé sur le réseau régulier le plus proche du vecteur de code. En fonction des modes de réalisation et des types de vecteurs, d'autres critères que la distance entre les vecteurs peuvent être retenus pour déterminer le vecteur voisin d'un vecteur de code.

Ainsi, l'étape 30 délivre un nouveau dictionnaire C' obtenu en arrondissant le dictionnaire CITER obtenu à l'étape 10, sur le réseau régulier.

L'arrondi des mots de code sur le réseau régulier est réalisé de la manière suivante, en considérant p le pas du réseau :

$$y'_i[k] = \left[ \frac{y_i[k]}{p} \right] \times p$$

où  $i = 1 \dots M$ ,  $k = 1 \dots n$  et  $[ \ ]$  indique l'arrondi à l'entier le plus proche.

Le procédé comprend ensuite une étape 32 d'élimination des doublons parmi les vecteurs de code modifiés formant le dictionnaire C'.

L'étape 32 comprend une sous-étape de recherche des doublons effectuée sur l'ensemble du dictionnaire C' en comparant chacun des vecteurs

de code aux autres vecteurs de code du dictionnaire puis une sous-étape de suppression de doublons.

A l'issue de cette étape 32, le dictionnaire C' peut comprendre moins de vecteurs de code que le dictionnaire CITER.

5 Le procédé comporte ensuite une étape 34 de fusion du dictionnaire modifié avec les vecteurs d'entraînement arrondis afin de compléter le dictionnaire si sa taille a été réduite.

10 Cette étape 34 débute par une sous-étape 36 de remplacement de chaque vecteur d'entraînement par un vecteur voisin, dit vecteur d'entraînement arrondi, choisi sur un réseau régulier de points. Avantageusement, le même réseau que celui de l'étape 30 est utilisé. Les vecteurs d'entraînement arrondis sont ensuite ordonnés selon leur fréquence d'apparition dans un ordre décroissant, avantageusement après avoir éliminé les doublons.

15 Le procédé comporte ensuite une sous-étape 38 de combinaison du dictionnaire C' de vecteurs de code modifiés avec les vecteurs d'entraînement arrondis. Au cours de cette sous-étape, le dictionnaire C' est complété par l'adjonction des vecteurs d'entraînement arrondis non présents dans le dictionnaire, avantageusement en suivant l'ordre des fréquences d'apparition  
20 décroissantes.

Ainsi, le dictionnaire C'' est construit de telle sorte qu'une distance minimale soit garantie entre les centroïdes formant les vecteurs de code du dictionnaire de sorte qu'il est qualifié de contraint. Dans l'exemple, les paramètres à quantifier sont pris sur une échelle logarithmique, un pas  
25 constant dans ce domaine équivaut à un pas logarithmique dans le domaine des énergies, ce qui correspond à la sensibilité de l'oreille humaine. En choisissant le pas p en fonction d'un seuil perceptuel, il est donc possible, grâce au procédé de l'invention, de générer un dictionnaire de quantification vectorielle statistique prenant en compte les seuils perceptuels.

30 De plus, en combinant le dictionnaire avec les vecteurs d'entraînement arrondis, une meilleure répartition des vecteurs de code est obtenue.

La figure 3 représente le dictionnaire C'' sous la forme d'un nuage de points en deux dimensions. On remarque, sur la figure 3, que le dictionnaire C'' obtenu suivant la méthode décrite précédemment, couvre une zone importante des vecteurs d'entraînement et présente une distance minimale entre les  
5 vecteurs de code deux à deux.

Le nombre de points apparents du dictionnaire sur la figure 3 est inférieur à celui de la figure 2 même si les deux dictionnaires ont en fait le même nombre de vecteurs de code. En effet les figures 2 et 3 ne représentent que la projection en deux dimensions de vecteurs de dimension plus  
10 importante. Les vecteurs de code du dictionnaire illustré à la figure 3 étant arrondis sur un réseau régulier, plusieurs vecteurs sont réduits à un seul point, ce qui n'est pas le cas pour le dictionnaire illustré à la figure 2.

Avantageusement, le procédé comporte en outre une phase  
15 d'optimisation du dictionnaire C'' dont un organigramme est représenté en référence à la figure 4.

L'optimisation débute avec la réception du dictionnaire C'' tel qu'obtenu à l'issue du procédé décrit en référence aux figures 1 à 3.

Cette phase d'optimisation consiste, dans le mode de réalisation décrit, à reprendre les mêmes étapes de traitement en les appliquant sur les vecteurs  
20 de code du dictionnaire C''. Toutefois, dans la phase d'optimisation décrite, à chaque itération, la classification, la détermination des centroïdes, la modification des vecteurs de code et l'élimination des doublons sont réalisées.

Ainsi, la phase 40 débute par une étape 42 de classification appliquée sur les vecteurs de code du dictionnaire C'' suivie d'une étape 44 de  
25 détermination des centroïdes. Ces étapes 42 et 44 sont similaires aux étapes 16 et 18 décrites précédemment.

Par la suite, une étape 46 de modification par arrondi sur le réseau régulier est réalisée. Cette étape 46 est suivie d'une étape 48 de remplacement des doublons.

30 Enfin, la phase d'optimisation comprend une étape 50 de mesure de la distorsion et un test 52 pour déterminer si la distorsion est inférieure à un seuil de tolérance ou si un nombre maximal d'itérations a été atteint.

Une telle phase d'optimisation permet d'améliorer encore la qualité du dictionnaire.

Bien entendu, d'autres modes de réalisation sont possibles.

En variante, le dictionnaire initial est créé par élagage des vecteurs d'entraînement arrondis sur le réseau régulier. C'est-à-dire que la création de ce dictionnaire initial comprend le remplacement des vecteurs d'entraînement par des vecteurs voisins choisis sur le réseau régulier puis l'élimination sélective, en fonction de leur fréquence d'apparition, des vecteurs d'entraînement arrondis.

10 Dans encore une autre variante, la phase d'optimisation consiste à répéter le procédé de la figure 1 à l'identique en utilisant comme dictionnaire initial le dictionnaire obtenu précédemment.

Dans une autre variante, l'étape de fusion est mise en œuvre entre les vecteurs de code modifiés et les vecteurs d'entraînement non arrondis. Dans un tel mode de réalisation, les vecteurs d'entraînement utilisés sont toutefois choisis afin de garantir le maintien de la distance minimale entre les vecteurs de code du dictionnaire.

Le procédé de génération d'un dictionnaire tel que décrit précédemment peut être mis en œuvre grâce à un programme pour tout type de calculateur et d'ordinateur.

En référence aux figures 5 et 6, un codeur et un décodeur utilisant des dictionnaires obtenus selon le procédé de l'invention sont décrits.

Ces codeurs et décodeurs font partie d'un système global de codage et décodage audio hiérarchique en sous-bandes qui fonctionne à trois débits possibles : 8, 12 ou 13.65 Kbit/s. En pratique le codeur fonctionne toujours au débit maximal de 13.65 Kbit/s, tandis que le décodeur peut recevoir le cœur à 8 kbit/s, ainsi qu'une ou deux couches d'amélioration, respectivement à 12 ou 13.65 kbit/s.

Le codeur 60 représenté sur la figure 5, comporte deux voies recevant le signal d'entrée. La première voie comporte une unité 62 de filtrage apte à extraire une bande basse BF de 0 à 4000 Hz du signal. La seconde voie comporte une unité similaire 64 apte à extraire une bande haute HF s'étalant

de 4000 à 8000Hz. Ces unités 62 et 64 comportent par exemple, des filtres et des modules de décimation réalisés de manière classique.

Le signal de bande basse est ensuite codé selon un codage CELP classique tel qu'un codage CELP encastré de 8 à 12 Kbits/s dans une unité de  
5 codage 66.

Le signal de bande haute est codé notamment en utilisant une quantification vectorielle dans une unité 70. Dans le mode de réalisation décrit, l'unité 70 est une unité de codage paramétrique dans laquelle les paramètres codés sont des enveloppes temporelle et fréquentielle du signal correspondant  
10 respectivement à la valeur efficace obtenue par moyenne quadratique (r.m.s. pour « root mean square ») par sous-trame ou sous-bande du signal de bande haute. Ces enveloppes sont passées dans un domaine logarithmique comme suit:

$$\sigma_{dB} = \frac{1}{2} \log_2 (\sigma)$$

15 où  $\sigma$  est la valeur de l'enveloppe.

Pour chaque trame du signal d'entrée des paramètres d'enveloppe temporelle et d'enveloppe fréquentielle sont extraits dans des modules d'analyse paramétrique 72 et 74. Ces paramètres dans le domaine logarithmique sont ensuite quantifiés conjointement dans un module 76 de  
20 quantification vectorielle en utilisant des dictionnaires générés selon l'invention.

Plus précisément, dans le mode de réalisation décrit, le codage des paramètres est réalisé par une quantification vectorielle dite produit cartésien à moyenne supprimée.

On calcule la moyenne  $\mu$  du vecteur d'enveloppe temporelle puis cette  
25 moyenne est quantifiée. La moyenne quantifiée est notée  $\mu_q$ . Les vecteurs d'enveloppe temporelle et d'enveloppe fréquentielle sont ensuite centrés sur  $\mu_q$  avant d'être quantifiés séparément et multiplexés.

La quantification de la moyenne  $\mu$  est réalisée par une quantification scalaire uniforme dont le pas  $p=0.99657$  est le seuil perceptuel, c'est-à-dire le  
30 pas du réseau régulier utilisé pour générer le dictionnaire.

La quantification vectorielle à produit cartésien est effectuée en utilisant le dictionnaire obtenu selon l'invention et dans lequel les vecteurs de code stockés sont des mots avec une distance minimale de  $p=0.99657$ , ce qui correspond à un pas de 6dB.

5 Les deux voies HF et BF sont ensuite multiplexées dans un multiplexeur 80 pour former le signal de sortie du codeur 60.

Le décodeur correspondant 100 est décrit en référence à la figure 6.

Ce décodeur comprend tout d'abord un démultiplexeur 102 séparant les voies correspondantes aux parties basse bande et haute bande du signal.

10 La voie correspondant au signal en basse bande est introduite dans un décodeur CELP 104. Ce décodeur CELP fournit également des paramètres d'excitation calculés de manière classique en soi.

La voie correspondant au signal en haute bande est introduite dans une unité de décodage 110 utilisant notamment une quantification vectorielle inverse 112.

La sortie de ce module 112 est fournie à un module de mise en forme temporelle 114. Ce module 114 reçoit également un signal d'excitation synthétique généré par un module 116 à partir des paramètres d'excitation CELP fournis par le décodeur CELP 104.

20 L'unité 110 comporte enfin un module de mise en forme fréquentielle 118.

De façon symétrique au codage des paramètres, le décodage s'effectue par dénormalisation des enveloppes temporelle et fréquentielle déquantifiées par la moyenne déquantifiée.

25 Enfin, les deux voies sont traitées dans des modules de traitement 120 et 122 avant d'être recombinaées entre elles dans un mixeur 124.

L'architecture et le fonctionnement d'un tel codeur et d'un tel décodeur sont classiques en eux-mêmes et ne seront pas décrits plus en détail ici. Une description détaillée est faite dans le document UIT-T, COM 16, D214 (WP 30 3/16), "High level description of the scalable 8-32 kbit/s algorithm submitted to the Qualification Test by Matsushita, Mindspeed and Siemens," Q.10/16, Study Period 2005-2008, Geneva, 26 July - 5 August 2005 (Source: Siemens).

## REVENDEICATIONS

1. Procédé de génération d'un dictionnaire de quantification vectorielle (C'') d'un signal, du type comportant une étape (10) d'analyse statistique de  
5 vecteurs d'entraînement représentatifs du signal déterminant un ensemble fini de vecteurs de code (CITER) représentant lesdits vecteurs d'entraînement, caractérisé en ce que le procédé comporte en outre une étape (30) de modification dudit ensemble fini de vecteurs de code pour imposer une distance minimale entre les vecteurs de code modifiés deux à deux, ces  
10 vecteurs de code modifiés formant ledit dictionnaire.

2. Procédé selon la revendication 1, caractérisé en ce que l'étape (10) d'analyse statistique comprend une sous-étape (12) de création d'un dictionnaire initial (C0), une sous-étape (16) de classification desdits vecteurs  
15 d'entraînement à partir du dictionnaire initial pour former des régions de quantification et une sous-étape (18) de détermination d'un centroïde pour chaque région, lesdits centroïdes étant lesdits vecteurs de code.

3. Procédé selon la revendication 2, caractérisé en ce que ladite sous-  
20 étape (12) de création d'un dictionnaire initial est réalisée à partir desdits vecteurs d'entraînement et comprend le remplacement de chaque vecteur d'entraînement par un vecteur arrondi choisi sur un réseau régulier de points et l'élimination sélective de vecteurs arrondis en fonction de leur fréquence d'apparition.

25

4. Procédé selon l'une quelconque des revendications 1 à 3, caractérisé en ce que l'étape d'analyse statistique comporte en outre une sous-  
étape (20) de calcul de la distorsion du dictionnaire initial et une sous-étape  
(22) de comparaison de cette distorsion avec un seuil de tolérance, ladite étape  
30 (10) d'analyse statistique étant répétée si ladite comparaison est négative.

5. Procédé selon l'une quelconque des revendications 1 à 4, caractérisé en ce que l'étape (30) de modification comprend le remplacement de chacun desdits vecteurs de code de l'ensemble fini par un vecteur de code voisin choisi sur un réseau régulier de points.

5

6. Procédé selon l'une quelconque des revendications 1 à 5, caractérisé en ce qu'il comporte en outre une étape (32) d'élimination des doublons parmi lesdits vecteurs de code modifiés.

10

7. Procédé selon la revendication 6, caractérisé en ce que ladite étape (32) d'élimination des doublons comporte une sous-étape de recherche des doublons par comparaison des vecteurs de code modifiés entre eux et une sous-étape de suppression des doublons en fonction de leur fréquence d'apparition.

15

8. Procédé selon l'une quelconque des revendications 1 à 7, caractérisé en ce qu'il comporte en outre une étape (34) de fusion des vecteurs de code modifiés avec des vecteurs d'entraînement présentant au moins ladite distance minimale entre eux et avec lesdits vecteurs de code modifiés.

20

9. Procédé selon la revendication 8, caractérisé en ce que l'étape fusion comporte une sous-étape (36) de remplacement de chaque vecteur d'entraînement par un vecteur arrondi choisi sur un réseau régulier de points et une sous-étape (38) de combinaison des vecteurs de code modifiés et des vecteurs d'entraînement arrondis.

25

10. Procédé selon la revendication 9, caractérisé en ce que ladite étape (36) de fusion comprend en outre l'ordonnancement des vecteurs d'entraînement arrondis selon leur fréquence d'apparition et la sélection d'un nombre fini de ces vecteurs pour les combiner avec les vecteurs de code modifiés.

30

11. Procédé selon l'une quelconque des revendications 9 et 10, caractérisé en ce que ladite sous-étape (38) de combinaison comprend l'adjonction auxdits vecteurs de code modifiés d'un nombre fini de vecteurs d'entraînement arrondis distincts desdits vecteurs de code modifiés.

5

12. Procédé selon l'une quelconque des revendications 3, 5 ou 9, caractérisé en ce que ledit réseau régulier de points a un pas correspondant à un niveau de sensibilité d'un récepteur destinataire dudit signal.

10

13. Procédé selon l'une quelconque des revendications 1 à 12, caractérisé en ce qu'il comporte en outre une phase (40) d'optimisation dudit dictionnaire formé desdits vecteurs de code modifiés.

15

14. Procédé selon la revendication 13, caractérisé en ce que la phase (40) d'optimisation comporte une étape (50) de calcul de la distorsion du dictionnaire et une étape (52) de comparaison de cette distorsion avec un seuil de tolérance, lesdites étapes de d'analyse statistique et de modification étant répétées si ladite comparaison est négative.

20

15. Procédé selon la revendication 13, caractérisé en ce que la phase (40) d'optimisation comporte au moins une itération du procédé mis en œuvre à partir du dictionnaire formé desdits vecteurs de code modifiés.

25

16. Dictionnaire de quantification vectorielle (C'') comportant des vecteurs de code représentant des vecteurs d'entraînement représentatifs d'un signal, caractérisé en ce qu'il est obtenu par un procédé selon l'une quelconque des revendications 1 à 15.

30

17. Programme d'ordinateur pour la génération d'un dictionnaire de quantification vectorielle (C'') d'un signal, caractérisé en ce qu'il comporte des instructions de code pour la mise en œuvre d'un procédé selon l'une

quelconque des revendications 1 à 15 lorsqu'il est exécuté sur un ordinateur d'un ordinateur.

5 18. Dispositif générateur d'un dictionnaire de quantification vectorielle (C''), caractérisé en ce qu'il comporte des moyens pour exécuter le programme d'ordinateur selon la revendication 17.

10 19. Procédé de codage d'un signal comportant au moins une étape d'analyse du signal pour en obtenir des paramètres représentatifs, caractérisé en ce qu'au moins un desdits paramètres représentatifs est codé par quantification vectorielle à l'aide d'un dictionnaire généré par un procédé selon l'une quelconque des revendications 1 à 15.

15 20. Codeur (60) de signaux comportant au moins des moyens (72, 74) d'analyse du signal pour en obtenir des paramètres représentatifs, caractérisé en ce qu'il comporte en outre des moyens (76) de codage d'au moins un desdits paramètre représentatifs par quantification vectorielle à l'aide d'un dictionnaire généré par un procédé selon l'une quelconque des revendications 1 à 15.

20

21. Procédé de décodage d'un signal comportant au moins une étape de traitement de paramètres représentatifs dudit signal, caractérisé en ce qu'au moins un desdits paramètres représentatifs est décodé par quantification vectorielle inverse à l'aide d'un dictionnaire généré par un procédé selon l'une  
25 quelconque des revendications 1 à 15.

30 22. Décodeur (100) de signaux comportant au moins des moyens de traitement de paramètres représentatifs dudit signal, caractérisé en ce qu'il comporte en outre des moyens (112) de décodage d'au moins un desdits paramètres représentatifs par quantification vectorielle inverse à l'aide d'un dictionnaire généré par un procédé selon l'une quelconque des revendications 1 à 15.

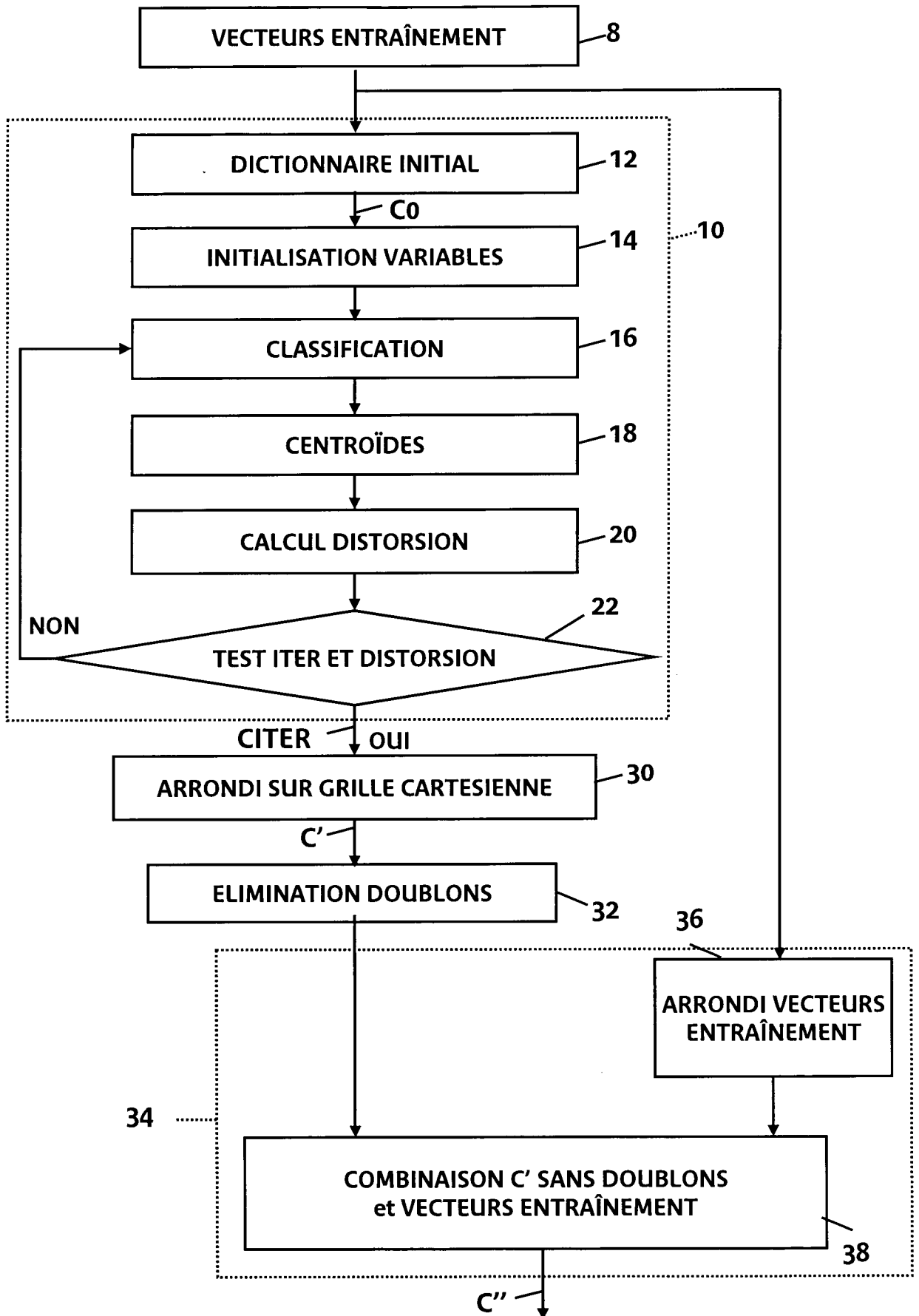
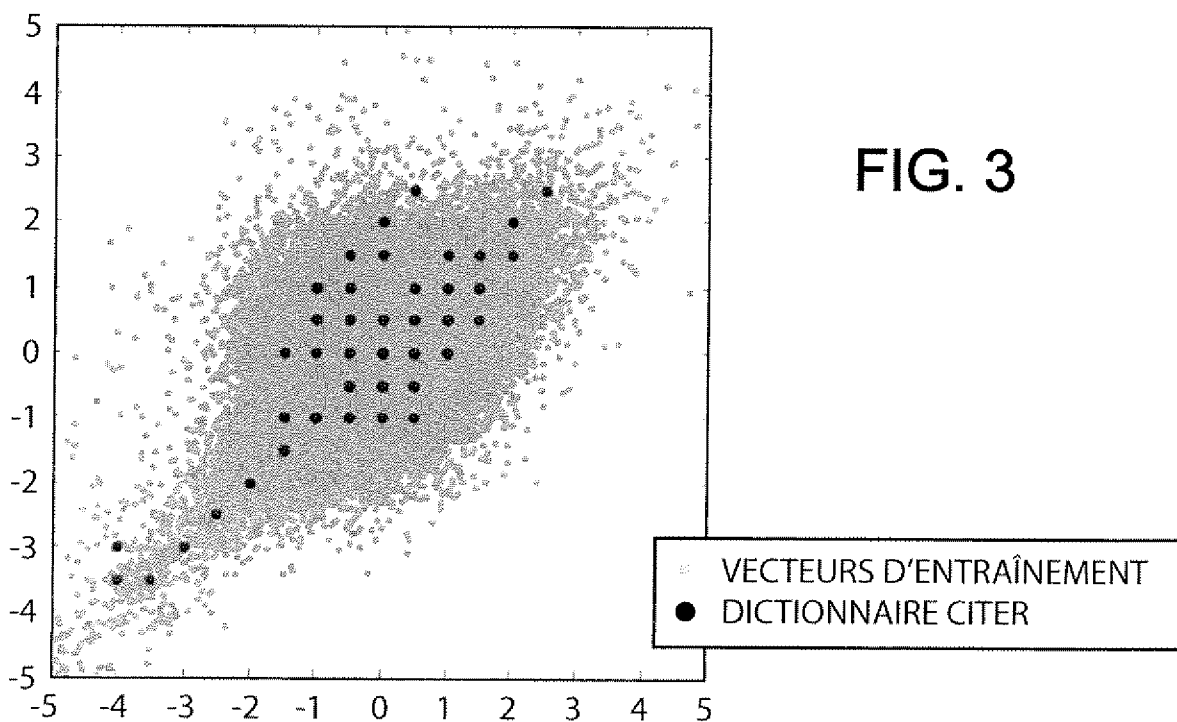
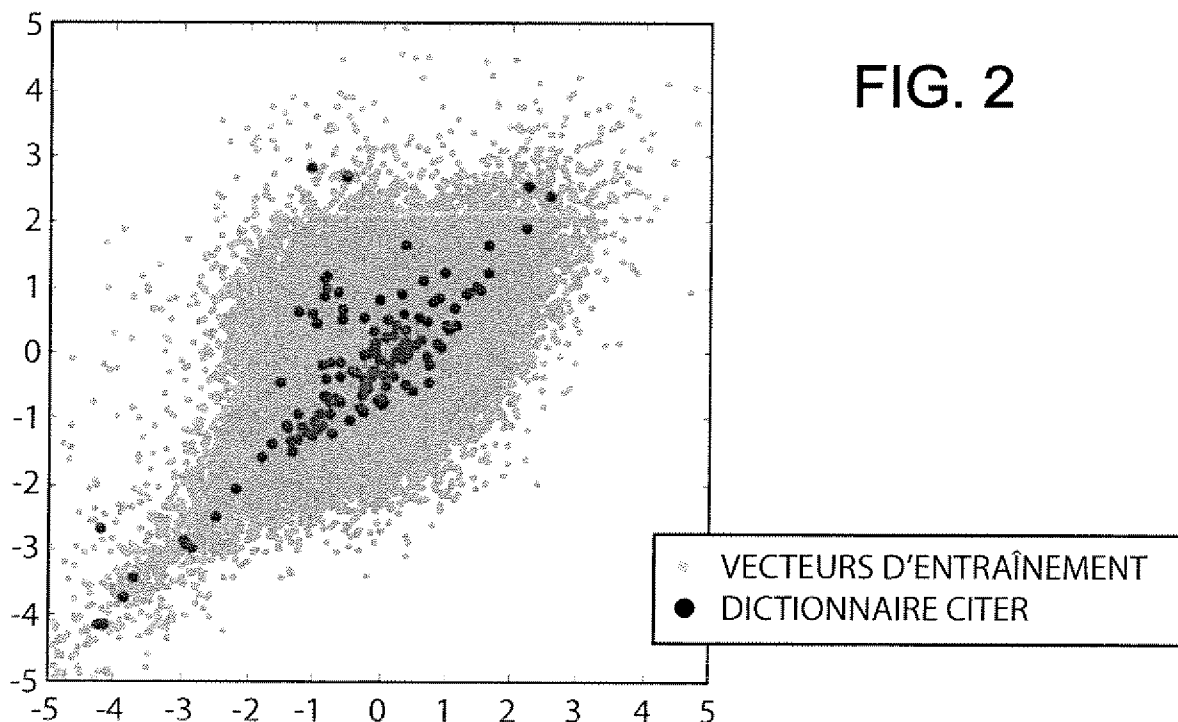


FIG 1



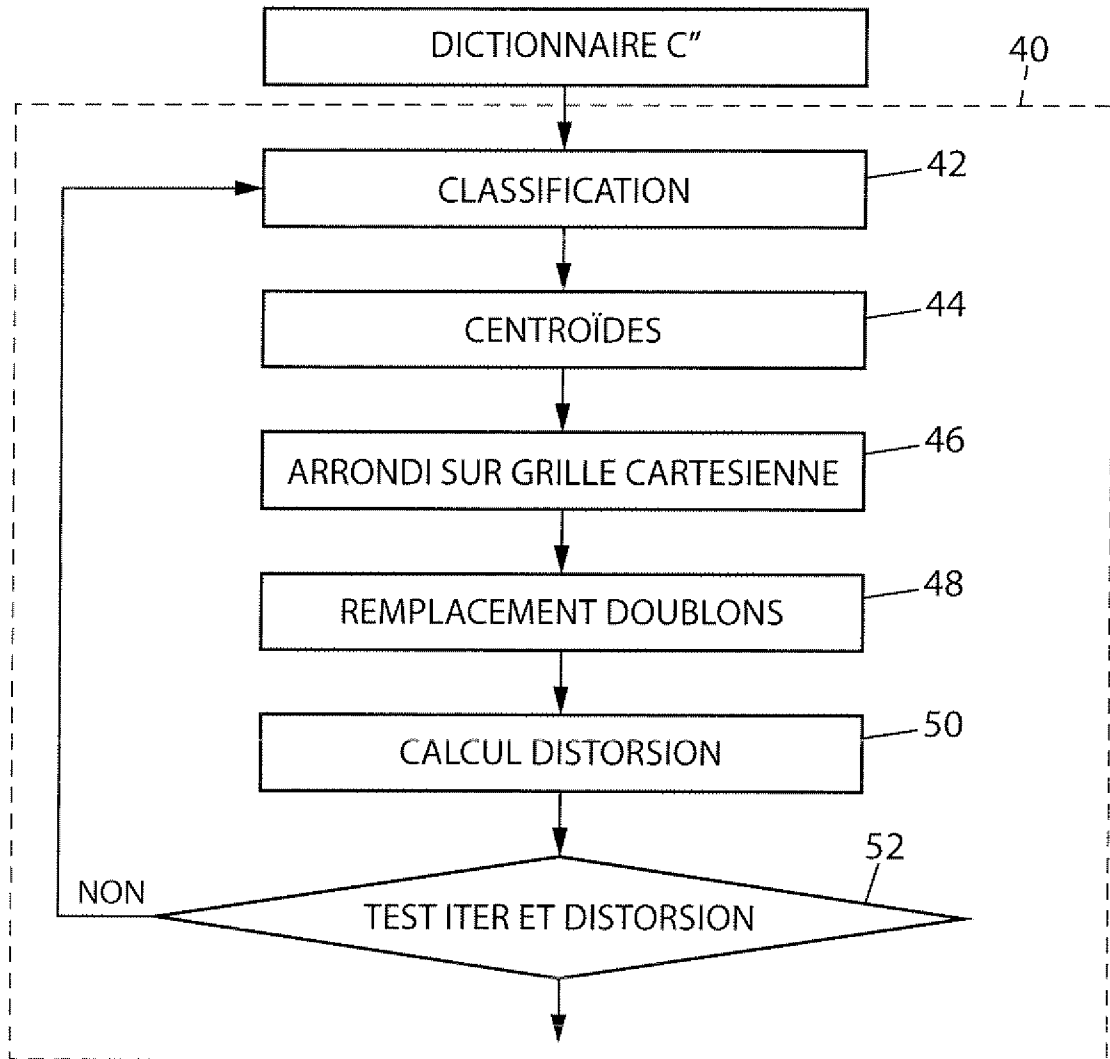


FIG. 4

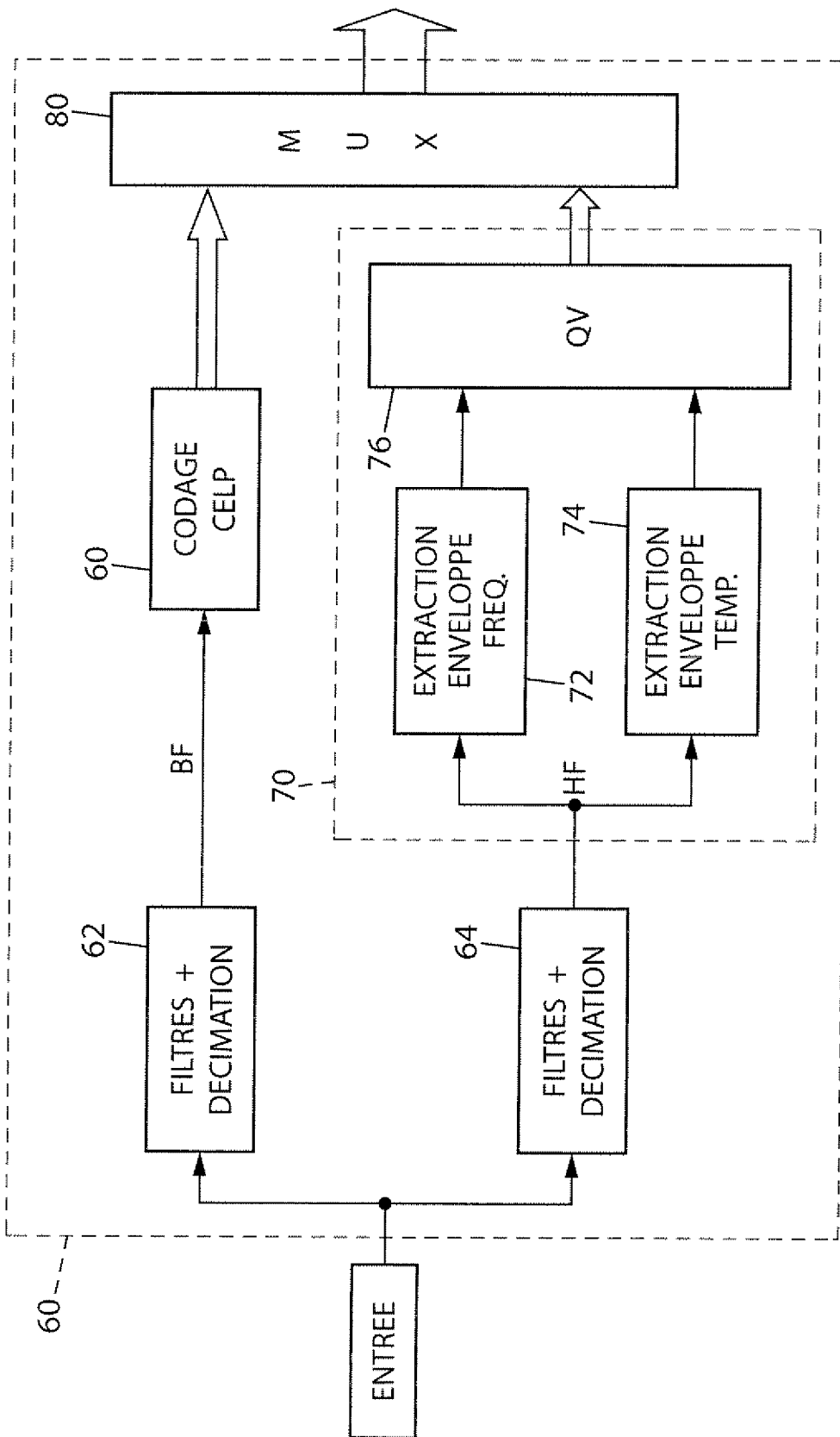


FIG. 5

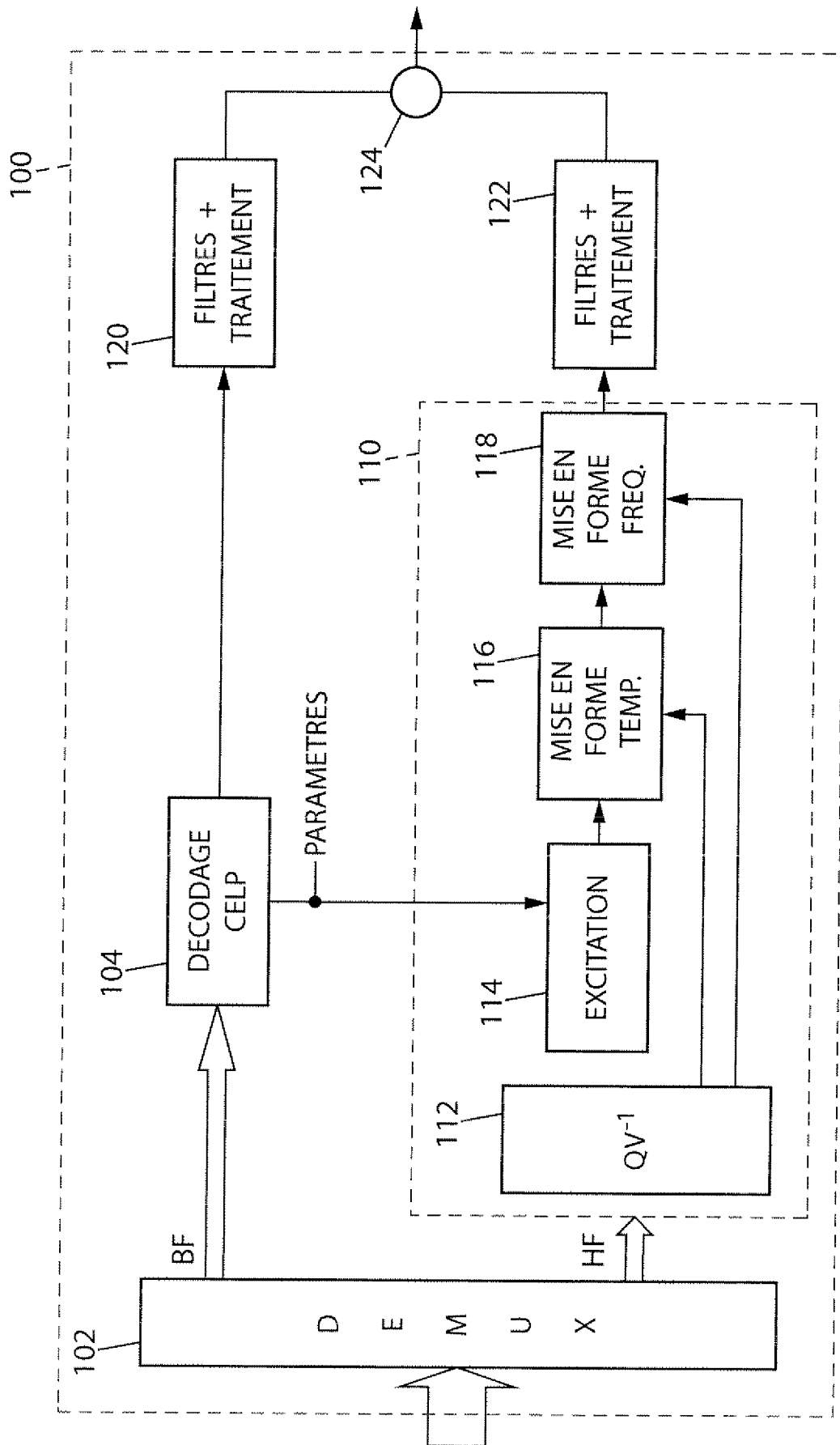


FIG. 6