(19) RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE

COURBEVOIE

(11) **N° de publication :** **3 139 212**

(à n'utiliser que pour les commandes de reproduction)

(21) **N° d'enregistrement national :** **22 08605**

(51) Int Cl[8] : ***H 04 N 19/119*** *(2024.01), G 06 F 18/231, G 06 F 16/20, H 04 N 19/96, G 06 N 20/00*

(12) **BREVET D'INVENTION** **B1**

(54) Système et procédé de construction de structures adaptatives de partitionnement de données en flux.

(22) **Date de dépôt :** 29.08.22.

(30) **Priorité :**

(43) **Date de mise à la disposition du public de la demande :** 01.03.24 Bulletin 24/09.

(45) **Date de la mise à disposition du public du brevet d'invention :** 04.10.24 Bulletin 24/40.

(56) **Liste des documents cités dans le rapport de recherche :**

Se reporter à la fin du présent fascicule

(60) **Références à d'autres documents nationaux apparentés :**

(○) **Demande(s) d'extension :**

(71) **Demandeur(s) :** *Commissariat à l'Energie Atomique et aux Energies Alternatives Etablissement public* — FR.

(72) **Inventeur(s) :** BLANCHART Pierre.

(73) **Titulaire(s) :** Commissariat à l'Energie Atomique et aux Energies Alternatives Etablissement public.

(74) **Mandataire(s) :** ATOUT PI LAPLACE.

FR 3 139 212 - B1

Description

Titre de l'invention : Système et procédé de construction de structures adaptatives de partitionnement de données en flux

Domaine technique

[0001] La présente invention concerne de manière générale le domaine du partitionnement de données, et en particulier un système et un procédé non-supervisé de construction de structures de partitionnement de données adaptatives à l'insertion de flux de données non-stationnaires.

[0002] Les grandes quantités de données accumulées dans de nombreux domaines permettent d'aborder des problèmes spécifiques comme la détection d'anomalies, la détection de changements ou encore la détection d'objets, grâce à des approches d'apprentissage telles que par exemple les approches d'apprentissage profond (ou « deep learning » en langue anglo-saxonne). Ces approches sont efficaces pour traiter certains des problèmes mentionnés. Cependant, elles ne passent pas à l'échelle sur des données massives et ne peuvent pas non plus s'adapter à des données se présentant sous forme de flux non-stationnaire, c'est-à-dire des données comprenant une distribution qui change au cours du temps du fait de l'arrivée de nouveaux groupes de données. Il est connu d'utiliser des systèmes de construction de structures de partitionnement afin de traiter au préalable des données massives et permettre de focaliser les traitements coûteux de certains processus d'apprentissage sur des ensembles de données plus réduits et stationnaires au cours du temps.

[0003] Les systèmes connus de construction de structures de partitionnement de données sont basés sur des procédés de partitionnement récursifs de distribution des données et en particulier des calculs, dans des espaces de grande dimension, d'axes optimisant certaines statistiques caractéristiques de multi-modalité des données. Les procédés de partitionnement reposent par exemple sur des expertises de longue date des statistiques d'ordre supérieur de type kurtosis. De tels systèmes permettent de générer des structures de partitionnement à partir d'un ensemble de donnée initial et comprenant des groupements (ou partitions) de données a priori cohérentes initialement (c'est-à-dire constitués d'éléments appartenant essentiellement à un seul groupe de données ayant une même étiquette).

[0004] Cependant, à l'insertion de nouvelles données (ou de flux de données) dans de telles structures initiales, la projection des données sur un axe de partitionnement peut être substantiellement altérée, modifiant la cohérence d'un ou plusieurs groupements de la structure. Ceci traduit en particulier la présence de nouvelles modalités (au sens de modalités statistiques) non anticipées (ou non présentes) dans les données initiales

utilisées pour construire la structure. Ces nouvelles modalités sont spécifiquement issues de ces nouvelles données insérées. Une telle situation correspond à une 'non-stationnarité' des données insérées par rapport aux données initiales.

[0005] Aujourd'hui, il n'existe aucun système ou procédé de construction de structures de partitionnement de données permettant de répondre aux problèmes d'insertion de données dans une structure de partitionnement initiale, les nouvelles données étant non-stationnaires par rapport au données de construction de cette structure.

[0006] Il existe ainsi un besoin pour un système et un procédé capable d'améliorer la construction de structures de partitionnement de données et en particulier l'insertion de flux de données non-stationnaires dans des structures de partitionnement afin de générer des partitions de données statistiquement et sémantiquement cohérents.

Résumé de l'invention

[0007] La présente invention vient améliorer la situation en proposant un dispositif de partitionnement de données apte à recevoir des données transmises par au moins une source de données, les données comprenant un ensemble de données initial B_0 et un ou plusieurs ensembles de données additionnels B_k. Le dispositif est configuré pour effectuer :

- au moins une opération de partitionnement initiale pour générer un arbre de partitionnement de données initial A_0 à partir de l'ensemble des données initial B_0, l'arbre étant associé à N sous-structures $P[n]$ comprenant chacune une distribution de données de construction $B_0[n]$ issue de l'ensemble de données initial B_0, et
- au moins une opération d'insertion pour insérer l'ensemble de données additionnels B_k dans l'arbre de partitionnement de données initial A_0, les sous-structures $P[n]$ comprenant en outre au moins une distribution de données insérée $B_k[n]$ issue de l'ensemble de données additionnels B_k.

[0008] Le dispositif est en outre configuré pour effectuer au niveau de chaque sous-structure $P[n]$ une comparaison $C[n]$ de la distribution de données de construction $B_0[n]$ avec la distribution de données insérée $B_k[n]$ et pour déterminer une valeur d'évaluation de la stationnarité des données insérées par rapport aux données de construction de l'arbre de partitionnement de données initial A_0 à partir de la comparaison. Le dispositif de partitionnement de données est également configuré pour mettre à jour le partitionnement de données de l'arbre à partir de la valeur d'évaluation.

[0009] Dans des modes de réalisation, l'opération de partitionnement initiale peut comprendre la détermination d'un axe de projection $V_0[n]$ de la distribution de données de construction $B_0[n]$ pour former au moins deux sous-ensembles de

données distincts issus de la distribution de données de construction $B_0[n]$. L'opération d'insertion de l'ensemble de données additionnels B_k peut être effectuée au niveau de chaque sous-structure $P[n]$ en prenant en compte l'axe de projection $V_0[n]$.

[0010] Selon certains modes de réalisation, la comparaison $C[n]$ peut comprendre une détermination d'une distance entre la projection de la distribution de données de construction $B_0[n]$ et la projection de la au moins une distribution de données insérée $B_k[n]$, la distance pouvant être par exemple la distance EMD.

[0011] Le dispositif peut être en outre configuré pour effectuer au moins une nouvelle opération de partitionnement, en réponse à une détection d'une condition de non-stationnarité des données. La nouvelle opération de partitionnement de données peut être effectuée au niveau d'au moins une sous-structure $P[n]$ à partir de la distribution de données $B_0[n]$ et de la au moins une distribution de données insérée $B_k[n]$ de manière à générer un arbre de partitionnement de données modifié A_k. La condition de non-stationnarité des données peut être vérifiée si la valeur d'évaluation de la stationnarité des données est supérieure à un seuil de non-stationnarité des données δ.

[0012] Avantageusement, la nouvelle opération de partitionnement peut comprendre la détermination d'un nouvel axe de projection $V_k[n]$ d'un ensemble de données de construction $S_{build}[n]$ pour former au moins deux nouveaux sous-ensembles de données distincts issus de l'ensemble de données de construction $S_{build}[n]$. L'ensemble de données de construction $S_{build}[n]$ peut comprendre la distribution de données de construction $B_0[n]$ et la au moins une distribution de données insérée $B_k[n]$.

[0013] Le seuil de non-stationnarité des données δ peut être préalablement fixé. La valeur d'évaluation de la stationnarité peut être égale à la valeur de comparaison des projection $C[n]$ telle que la valeur de comparaison satisfait la condition de non-stationnarité des données : $C[n] > \delta$.

[0014] Alternativement, Le seuil de non-stationnarité des données δ peut être ajusté en fonction d'une probabilité de fausse alarme fixée ζ. La détermination de la valeur d'évaluation de la stationnarité peut comprendre la mise en œuvre d'un test statistique basé sur une détection a contrario de non-stationnarités des données.

[0015] Dans certains modes de réalisation, la valeur d'évaluation de la stationnarité de la détection a contrario peut être déterminée à partir d'une fonction de répartition F_E définie à partir de la valeur de comparaison des projections $C[n]$, la condition de non-stationnarité des données étant telle que : $F_E(C[n]) = P(E > \delta | H0)$.

[0016] L'invention fournit également un système de construction de structures de partitionnement de données comprenant au moins une source de données, le dispositif de partitionnement de données et un module d'exploitation. Le module d'exploitation est configuré pour déterminer au moins un score de pureté moyenne associé à une distribution de données d'une sous-structure $P[n]$ et pour évaluer la qualité de l'arbre de partitionnement de données à partir du au moins un score de pureté moyenne déterminé.

[0017] L'invention fournit également un procédé mis en œuvre par un dispositif de partitionnement de données comprenant les étapes consistant à :

- Recevoir un ensemble de données initial B_0 depuis au moins une source de données ;
- Effectuer au moins une opération de partitionnement initiale pour générer un arbre de partitionnement de données initial A_0 à partir de l'ensemble des données initial B_0, l'arbre étant associé à N sous-structures $P[n]$ comprenant chacune une distribution de données de construction $B_0[n]$ issue de l'ensemble de données initial B_0 ;
- Recevoir un ou plusieurs ensembles de données additionnels B_k depuis au moins une source de données ;
- Effectuer au moins une opération d'insertion pour insérer l'ensemble de données additionnels B_k dans l'arbre de partitionnement de données initial A_0, les sous-structures $P[n]$ comprenant en outre au moins une distribution de données insérée $B_k[n]$ issue de l'ensemble de données additionnels B_k ;
- Effectuer au niveau de chaque sous-structure $P[n]$ une comparaison $C[n]$ de la distribution de données de construction $B_0[n]$ avec la distribution de données insérée $B_k[n]$ et déterminer une valeur d'évaluation de la stationnarité des données insérées par rapport aux données de construction de l'arbre de partitionnement de données initial A_0 à partir de la comparaison ; et
- Mettre à jour l'arbre de partitionnement de données à partir de la valeur d'évaluation.

[0018] Le système et le procédé incrémental d'identification de non-stationnarités selon les modes de réalisation de l'invention permettent d'améliorer la construction de structures de partitionnement de données et en particulier l'insertion de flux de données dans des structures de partitionnement.

[0019] Ils permettent notamment de générer une solution de mise à jour partielle de la structure et adaptive aux nouvelles modalités statistiques présentes dans les données insérées afin de former une structure finale comprenant des groupements (ou partitions) de données statistiquement et sémantiquement cohérents, garantissant ainsi des propriétés d'exploitation de données pour des applications à des processus

d'apprentissage.

Description des figures

[0020] D'autres caractéristiques, détails et avantages de l'invention ressortiront à la lecture de la description faite en référence aux dessins annexés donnés à titre d'exemple.

[0021] [Fig.1] La [Fig.1] est un schéma représentant une structure globale d'un système de construction de structures de partitionnement de données, selon des modes de réalisation de l'invention.

[0022] [Fig.2] La [Fig.2] représente deux exemples d'application sur une même image satellite, selon des modes de réalisation de l'invention.

[0023] [Fig.3] La [Fig.3] est un schéma représentant un arbre de partitionnement de données et un graphique illustrant la partition d'une distribution de données par un axe intelligent, selon des modes de réalisation de l'invention.

[0024] [Fig.4] La [Fig.4] est un schéma représentant un arbre de partitionnement de données et un graphique illustrant la partition de deux distributions de données par un axe intelligent, selon des modes de réalisation de l'invention.

[0025] [Fig.5] La [Fig.5] est un schéma représentant un arbre de partitionnement de données et un graphique illustrant la partition d'un ensemble de données contenant deux distributions de données par un axe intelligent, selon des modes de réalisation de l'invention.

[0026] [Fig.6] La [Fig.6] est un organigramme représentant un procédé de génération d'un arbre de partitionnement de données, selon des modes de réalisation de l'invention.

[0027] [Fig.7] La [Fig.7] est un organigramme représentant un procédé d'insertion de données et de comparaison des projections de données, selon des modes de réalisation de l'invention.

[0028] [Fig.8] La [Fig.8] est un organigramme représentant un procédé de mise à jour d'un arbre de partitionnement de données, selon des modes de réalisation de l'invention.

[0029] Des références identiques sont utilisées dans les figures pour désigner des éléments identiques ou analogues. Pour des raisons de clarté, les éléments représentés ne sont pas à l'échelle.

Description détaillée

[0030] La [Fig.1] représente schématiquement une structure globale d'un système 10 de construction de structures de partitionnement de données, selon certains modes de réalisation de l'invention.

[0031] Le système 10 de construction de structures de partitionnement de données comprend des sources de données 12 et un dispositif de partitionnement de données 100.

[0032] Les sources de données 12 comprennent un ou plusieurs ensembles de données pouvant être compris et/ou acquis par le système 10.

[0033] Dans des modes de réalisation, une source de données peut être un capteur d'acquisition de données associé à l'acquisition d'un ou plusieurs signaux mono- ou multidimensionnels, tels que des signaux acoustiques, des signaux visuels, des signaux associés à des grandeurs physiques (température, pression), etc.

[0034] Selon certains modes de réalisation, une source de données peut comprendre une interface de réception (non représentée sur les figures) destinée à recevoir des données à travers différents systèmes annexes, tels que des capteurs d'acquisition de données par exemple.

[0035] Les données comprises dans les sources de données 12 peuvent donc être d'un ou plusieurs types. Par exemple, il peut s'agir d'images, de séquences audio et/ou de vidéos et/ou des données multimédia. Plus généralement, les données des sources de données 12 peuvent être tout type de données à catégoriser et peuvent être acquises par un capteur. Un ensemble de données peut faire référence à une grande quantité de données (c'est-à-dire de gros volumes de données encore appelés « données massives »), correspondant par exemple à plusieurs téraoctets de données.

[0036] En outre, les données peuvent être réceptionnées (ou considérées à traiter) en flux (ou en séries temporelles) de manière continue et/ou discontinue. Les flux de données peuvent être stationnaires et/ou non-stationnaires.

[0037] Tel qu'utilisé ici, le terme « non-stationnarité » fait référence à une série temporelle de données vérifiant des principes de non-stationnarité déterministe. De façon similaire, le terme « stationnarité » peut être associé à une série temporelle de données vérifiant une propriété dite de « stationnarité forte », c'est-à-dire pour laquelle la distribution des données ne change pas avec le temps et l'arrivée de nouvelle données. Mathématiquement, une série temporelle de données $\{Z(1),\ Z(2),\ Z(t)\}$ est dite stationnaire au sens fort si pour toute fonction f mesurable, les termes $f(Z(1),\ Z(2),\ Z(t))$ et $f(Z(1+k),\ Z(2+k),\ Z(t+k))$ vérifient une même loi.

[0038] Le dispositif de partitionnement de données 100 est configuré pour effectuer une opération de partitionnement de données, c'est-à-dire pour partitionner (ou scinder) une partie et/ou l'ensemble des données compris dans les sources de données 12 en un certain nombre de sous-ensembles de données (appelés aussi 'groupements' ou 'partitions').

[0039] Le système 10 de construction de structures de partitionnement de données (et donc le dispositif 100) peut être utilisé comme un des processeurs de données mis en œuvre dans un système de traitement de données provenant d'un système antennaire et/ou d'imagerie par exemple (non représenté sur les figures) monté à bord d'un satellite ou d'une station d'acquisition de données et de contrôle sol, pour des missions spatiales,

météorologiques et/ou scientifiques.

[0040] Par exemple et sans limitation, les sources de données 12 peuvent comprendre une image définie par un certain nombre de pixels ou groupements de pixels (appelé également « patches »). Le découpage d'une image en patches peut être effectué à partir d'un procédé de fenêtre glissante. Les données peuvent être par exemple des paquets d'informations brutes ou d'informations prétraitées (c'est-à-dire dérivées de données brutes, encore appelées « descripteurs »). Des descripteurs peuvent être des caractéristiques, de forme ou de texture par exemple, extraites généralement de patches. Des informations brutes peuvent être des valeurs colorimétriques (selon le code RGB ou le code LAB notamment) associées chacune à un pixel de l'image. Le dispositif de partitionnement de données 100 peut ainsi être configuré pour partitionner l'image en différentes catégories de pixels ou de patches associées chacune à une étiquette (ou « label » en langue anglo-saxonne), selon un ou plusieurs critères prédéfinis. La [Fig.2] illustre un tel exemple de réalisation de l'invention. Les éléments (a) et (b) de la [Fig.2] représentent une même image satellite, comprenant des zones de forêt et des zones de champs dont les pixels ou les patches ont été regroupés, suite à une opération de partitionnement de données, respectivement en une première catégorie de pixels (zones z1 en gris clair sur la [Fig.2](a) associées à une étiquette 'zones de forêt') et en une deuxième catégorie de pixels (zones z2 en gris foncé sur la [Fig.2](b) associées à une étiquette 'zones de champs'). Dans l'exemple de l'image satellite présenté sur la [Fig.2], les critères de partitionnement des pixels peuvent être définis à partir des informations colorimétriques de l'image initiale (par exemple les 'zones de forêt' sont associées à un code RGB compris dans intervalle représentant le « vert foncé » et les 'zones de champs' sont associées à un code RGB compris dans intervalle représentant le « jaune »).

[0041] Le dispositif de partitionnement de données 100 peut par exemple comprendre un module de traitement 140 et un module de sauvegarde 160.

[0042] Une opération de partitionnement de données peut être effectuée par le dispositif de partitionnement de données de manière récursive (ou itérative) sur des ensembles et sous-ensembles des données compris dans les sources de données 12, de manière prédéfinie et/ou en réponse à une commande émise par le module de traitement 140 par exemple.

[0043] Le dispositif de partitionnement de données 100 est alors configuré pour générer un index, à partir d'une ou de plusieurs opérations de partitionnement de données.

[0044] Tel qu'utilisé ici, le terme d'« index » (ou index de données) fait référence un structure de partitionnement, appelée aussi arbre de partitionnement. Une fonctionnalité de l'indexation de données est, par exemple, de permettre la sélection rapide de un ou plusieurs sous-ensembles de données, d'un ensemble de données, associés à

une nouvelle donnée injectée (ou insérée) dans cet ensemble de données.

[0045] Ainsi, le dispositif de partitionnement de données 100 est configuré pour effectuer une opération d'insertion de données, c'est-à-dire pour insérer, dans un index initial noté A_0 préalablement généré à partir d'un ensemble des données initial noté B_0, un ou plusieurs ensembles de données additionnels notés B_k tel que $k \in [1, K]$ et $K \geq 1$. Le ou les ensembles de données additionnels B_k sont réceptionnées (ou considérées à traiter) en flux dans les sources de données 12.

[0046] L'index initial A_0 comprend un certain nombre $(N-1)$ de sous-ensembles de données, issues d'opérations de partitionnement effectuées à partir de l'ensemble des données initial B_0. Ainsi, l'index initial A_0 comprend N sous-structures, encore appelées « nœuds » et notées $P[n]$ tel que $n \in [1, N]$ et $N > 1$. Chaque nœud $P[n]$ de l'index initial A_0 comprend une partition de données, notée $B_0[n]$ et associée à une distribution de données au sens statistique (comprenant par exemple des parties et/ou l'ensemble des données à partitionner). La partition de données $B_0[n]$ est issue de l'ensemble des données initial B_0 et utilisée pour construire une partie de l'index initial A_0. L'ensemble des données initial est par exemple noté $B_0[1]$. L'ensemble des données initial B_0 et/ou les partitions (ou distributions) de données $B_0[n]$ sont encore appelées ci-après 'données historiques de construction de l'index initial A_0'.

[0047] En réponse à l'insertion de chaque ensemble de données additionnel B_k, chaque nœud $P[n]$ d'un index modifié noté A_k peut être associé à deux ensembles de données. Le premier ensemble de données est encore appelé 'ensemble de données de construction', noté $S_{build}[n]$, et comprend au moins une distribution de données $B_0[n]$. Le deuxième ensemble de données est encore appelé 'ensemble de données de insérées', noté $S_{insert}[n]$, et comprend au moins une partition de données insérées $B_k[n]$, issue d'un ensemble de données additionnel B_k.

[0048] Le dispositif de partitionnement de données 100 peut être en outre configuré pour comparer au niveau de chaque nœud $P[n]$ les distributions de données $B_0[n]$ et $B_k[n]$, et pour évaluer chacune des différentes comparaisons de l'index initial A_0 en fonction d'un seuil de 'non-stationnarité des données' noté δ.

[0049] Le dispositif de partitionnement de données 100 est configuré pour ensuite mettre à jour au moins partiellement l'index initial A_0, c'est-à-dire pour effectuer au moins une opération de partitionnement de données au niveau d'au moins un nœud $P[n]$ à partir d'un ensemble des distributions de données $\{B_0[n], \ldots B_k[n]\}$ répartis entre les deux ensembles de données $S_{build}[n]$ et $S_{insert}[n]$.

[0050] Un tel dispositif de partitionnement de données 100 permet ainsi de générer des structures de partitionnement de données qui sont adaptatives à l'insertion de flux de

données non-stationnaires.

[0051] Le dispositif de partitionnement de données 100 peut être configuré pour effectuer de manière récursive une opération de partitionnement sur les sous-ensembles des données engendrés par chacune des opérations de partitionnement à partir d'au moins un nœud $P[n]$.

[0052] Le dispositif de partitionnement de données 100 (via par exemple le module de traitement 140) est ainsi apte à générer un index modifié A_k, en fonction de la ou des opérations de partitionnement des données effectuées à partir de la mise à jour d'au moins un nœud $P[n]$. La génération de l'index modifié A_k permet notamment une adaptation aux nouvelles modalités statistiques présentes dans les données insérées B_k.

[0053] La figure 3(a) représente schématiquement un exemple d'index A_0 et plus précisément un arbre de partitionnement binaire, c'est-à-dire basé sur un découpage en deux, de manière récursive, de distributions de données à partir d'un ensemble de données initial B_0.

[0054] Les figures 4(a) et 5(a) représentent la modification de l'index initial A_0 suite à l'insertion des ensembles de données additionnels B_k selon certains modes de réalisation de l'invention.

[0055] Comme illustré sur les figures 3(a), 4(a) et 5(a), dans une structure de partitionnement binaire, les nœuds peuvent également être notés $P[i, j]$. Selon cette notation, i fait référence à un niveau de profondeur de l'arbre de partitionnement binaire et j fait référence à un degré de largeur de l'arbre. La profondeur maximale de l'arbre binaire peut être notée L tel que $i \in [0, L-1]$, et $j \in [1, 2^i]$ pour chaque niveau i. De plus, chaque partition de données, notée $B[i, j]$, d'un nœud $P[i, j]$ génère deux nœuds appelés aussi 'nœud fils', par exemple $P[(i+1), (2j-1)]$ et $P[(i+1), (2j)]$. Un nœud générant d'autres nœuds est appelé 'nœud parent'. Les nœud fils sont alors associés chacun à une distribution de données (pouvant être partitionnées), respectivement $B[(i+1), (2j-1)]$ et $B[(i+1), (2j)]$. Ainsi, chaque niveau i de l'arbre binaire représente une partition de l'ensemble de données initial noté $B[0,1]$ tel qu'exprimé dans l'équation (01) suivante :

[0056] $$\forall i, \quad \bigcup_{j=1}^{2^i} B[i, j] = B[0,1] \quad (01)$$

[0057] Un nœud comprenant une distribution de données n'ayant pas subi d'opération de partitionnement, désigne un nœud « feuille ».

[0058] Comme illustré sur la [Fig.1], le module de traitement 140 du dispositif de partitionnement de données 100 peut comprendre une unité de projection 142 configurée pour mettre en œuvre la ou les opérations de partitionnement.

[0059] En particulier, l'unité de projection 142 peut être configurée pour déterminer le ou

les axes de projection possibles associés respectivement à des parties et/ou l'ensemble des données à partitionner. Les axes de projection mettant en évidence différentes distributions de données intrinsèques aux sources de données 12 sont appelés 'axes intelligents'. L'unité de projection 142 est alors configurée pour choisir (ou rechercher par itération par exemple) le ou les axes intelligents associés à une ou plusieurs distribution des données parmi au moins certains axes de projection possibles déterminés préalablement par exemple.

[0060] Il est à noter que dans des espaces de données de grande dimension, les axes intelligents choisis pour générer un index ont pour but d'optimiser certaines statistiques caractéristiques de la multi-modalité des données dans ces espaces et de garantir la cohérence sémantique des différents sous-ensembles de données de cet index. Les axes intelligents choisis permettent d'effectuer des découpages de données dans des zones dites « zones de faible densité de données » situées entre les modes et permettent de garantir l'intégrité d'ensembles denses de données.

[0061] Dans des modes de réalisation, les opérations de partitionnement des données peuvent reposer sur des statistiques d'ordre supérieur de type kurtosis (moment d'ordre 4). Le kurtosis (encore appelé « coefficient d'acuité » ou « coefficient d'aplatissement ») désigne une mesure réalisée sur une distribution de données, utilisée notamment en théorie des probabilités et en statistique, représentant une répartition des données par rapport à une 'distribution normale' vérifiant une loi normale.

[0062] En particulier, un axe intelligent (caractérisé par exemple par un vecteur directeur) peut être obtenu en minimisant le calcul du kurtosis de la projection d'une distribution des données sur cet axe (la minimisation est effectuée par rapport à d'autres projections sur d'autres axes déterminés). Par définition et du fait des propriétés du kurtosis, une projection binaire d'une distribution des données a une valeur minimale (par exemple proche de un, le kurtosis étant supérieur à 1) et deux ou plusieurs « bosses » ou « pics », encore appelées 'modes', de part et d'autre d'une ou plusieurs valeurs centrales prédéterminées (par exemple zéro). En particulier, le procédé de minimisation du kurtosis de données centrées réduites peut consister à déterminer l'axe suivant lequel la distribution des données est la plus bimodale possible, avec deux modes situés de part et d'autre d'une valeur centrale (0 dans le cas de données centrées). À l'extrême, un kurtosis de 1 sur des données centrées réduites correspond à une distribution de données composées de deux Dirac situées en -1 et 1. En raison de cette propriété, le critère de minimum de kurtosis est particulièrement adapté au procédé de partitionnement binaire.

[0063] Les arbres de partitionnement binaire sont associés à une classe générique d'algorithmes de partitionnement générant une indexation multi-résolution d'un jeu de données. Chaque nœud $P[i, j]$ d'un arbre de partitionnement binaire peut posséder

deux nœuds fils, appelés nœud 'gauche' et nœud 'droit' conventionnellement, qui résultent de la partition binaire d'un ensemble ou sous-ensemble de données par un hyperplan séparateur, ou de façon équivalente suivant les coordonnées de projection des points sur un axe normal à cet hyperplan.

[0064] Un exemple de projection binaire d'une distribution des données sur un axe intelligent est représenté par l'histogramme 1D de la figure 3(b) composé d'un mode 'gauche' et d'un mode 'droit', et présentant un 'creux' en 'zéro'. Chacun des modes d'une projection de distributions des données sur un axe intelligent correspond à la projection d'une partition de données (i.e. sous-ensemble de données). L'arbre de partitionnement binaire A_0 sur la figure 3(a) est par exemple initialisé selon un découpage en deux de l'ensemble de données initial B_0 pour former les deux sous-ensembles de données $B_0[1,1]$ et $B_0[1,2]$ tels que représentés sur la figure 3(b) illustrant une représentation de la distribution de données B_0 (c'est-à-dire la distribution), selon les coordonnées de l'axe intelligent appliqué. Par exemple et sans limitation, cette représentation (aussi notée H_{build}) peut être un histogramme 1D normalisé.

[0065] Selon certains modes de réalisation, les axes intelligents et autres paramètres de partitionnement déterminés par l'unité de projection 142 peuvent être issus de méthodes de partitionnement classiques. Les arbres de partitionnement peuvent être par exemple et sans limitation :

– des arbres dits « arbres k-d » (abréviation d'arbre à k dimensions) dont la méthode de partitionnement permet d'effectuer des partitions binaires suivant les différents axes de coordonnées des données ;

– des arbres dits « arbres PCA » (abréviation d'arbre à analyse des composants principaux) dont la méthode de partitionnement permet de partitionner les données en chaque nœud suivant le premier axe principal d'un sous-ensemble de données correspondant à ce nœud, comme décrit dans les articles "Refinements to nearest-neighbor searching ink-dimensional trees" de R. F. Sproull 1991, Algorithmica, 6(1-6):579–589, et "A fast nearest-neighbor algorithm based on a principal axis search tree" de J. McNames 2001, IEEE Transactions on pattern analysis and machine intelligence, 23(9):964–976 ;

– des arbres dits « arbres 2Means » (ou « arbres 2M ») dont la méthode de partitionnement utilise un axe obtenu en joignant les centres de deux clusters résultant du regroupement (ou 'clustering') binaire d'un sous-ensemble de données correspondant à un nœud, comme décrit dans les articles "A branch and bound algorithm for computing k-nearest neighbors" de K. Fukunaga and P. M. Narendra 1975, IEEE transactions on computers, 100(7):750–753, 1975, et "Scalable recognition with a vocabulary tree" de D. Nister and H. Stewenius 2006, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), volume 2, pages 2161–2168. Les arbres

2Means peuvent notamment être utilisés en imagerie satellitaire pour effectuer des clustering hiérarchiques appelés « dyadic-kmeans clustering », comme décrit dans l'article "Cluster structure evaluation of dyadic k-means algorithm for mining large image archives" de H. Daschiel and M. P. Datcu 2003, In Image and Signal Processing for Remote Sensing VIII, volume 4885, pages 120–130. International Society for Optics and Photonics ;

– des arbres dits « arbres random projection » dont la méthode de partitionnement utilise la projection des données sur un ou plusieurs axes aléatoires et dont l'optimisation de la structure porte alors sur la détermination d'un seuil de « découpe intelligent », comme décrit dans l'article "Random projection trees and low dimensional manifolds" de S. Dasgupta and Y. Freund 2008, In Proceedings of the fortieth annual ACM symposium on Theory of computing, pages 537–546 ;

– des arbres dits « arbres Maximum-Margin » dont la méthode de partitionnement utilise la séparation des données en chaque nœud suivant un hyperplan de marge maximale calculé en utilisant un algorithme de « maximum margin clustering », décrit dans l'article "Maximum margin clustering" de L. Xu, J. Neufeld, B. Larson, and D. Schuurmans 2005, In Advances in neural information processing systems, pages 1537–1544 ; et

– d'autres arbres dont la méthode de partitionnement autorise par exemple certains sous-ensembles de nœuds fils à posséder des regroupements de données de recouvrement, comme décrit dans l'article "An investigation of practical approximate nearest neighbor algorithms" de T. Liu, A. W. Moore, K. Yang, and A. G. Gray 2005, In Advances in neural information processing systems, pages 825–832.

[0066] Dans d'autres modes de réalisation, les axes intelligents déterminés par l'unité de projection 142 peuvent être des axes dits « axes mini-kurtiques » pour lesquels l'unité de projection 142 peut utiliser une méthode de partitionnement comprenant différentes étapes. Ces étapes peuvent comprendre les étapes consistant à :

– Acquérir plusieurs jeux de données, par exemple des images, et les organiser dans une matrice de données ayant une première dimension égale à la dimension d'un jeu de données, par exemple la taille d'une image, et une seconde dimension égale au nombre d'acquisitions dans le jeu de données ;

– Centrer et ortho-normaliser la matrice de données, de manière à transformer la matrice de données en une matrice dont les composantes selon la seconde dimension ont une moyenne nulle, une norme unité et sont orthogonales deux à deux. Cette étapes peut être mise en œuvre au moyen d'une analyse en composantes principales par exemple appliquée de manière à réduire la première dimension de la matrice de données ;

– Rechercher au moins un axe de projection tel que la projection de la matrice de

données sur cet axe présente un kurtosis minimal ; cette recherche peut notamment être mise en œuvre en recherchant, pour au moins un couple de composantes de la matrice de données selon la seconde dimension, une combinaison linéaire de ce couple ayant un kurtosis minimal, par exemple, en recherchant une matrice de rotation de Givens, définie par un angle de rotation θ, pour laquelle le produit le couple de composantes et de cette matrice de rotation de Givens présente un kurtosis minimal pour l'une de ses deux composantes ou présente une différence maximale entre les kurtosis des deux composantes ; et

– Partitionner les données en au moins deux groupes ou groupements distincts de données en fonction du signe des composantes de la matrice de données projetées sur chaque axe de projection obtenu.

[0067] L'utilisation d'axes mini-kurtiques permet de modéliser de manière précise et compacte une distribution des données initiale. Par ailleurs, l'utilisation d'axes mini-kurtiques permet d'obtenir des arbres de partitionnement ayant des performances supérieures en termes de pureté moyenne des feuilles, lors d'insertion de données en flux, par rapport aux autres arbres de partitionnement connus de l'homme du métier. De plus, l'utilisation d'axes mini-kurtiques donne lieu à des mises à jour moins fréquentes de la structure d'index au fur et à mesure des itérations d'insertion de données en flux. Chaque axe mini-kurtique utilisé pour effectuer une partition binaire des données correspond à une direction dite « direction de bimodalité maximale » dans un espace des données.

[0068] Dans des modes de réalisation, la construction d'un arbre de partitionnement (binaire notamment) peut prendre en compte la profondeur L, ainsi que d'éventuels paramètres de construction de la procédure de calcul de l'axe de partitionnement. En particulier, le procédé de calcul de l'axe intelligent peut comprendre la mise en œuvre d'un axe mini-kurtique ou d'un des axes utilisés dans une méthode de partitionnement classique. Le procédé de calcul de l'axe intelligent peut également comprendre la mise en œuvre alternée de plusieurs types d'axes suivant un paramètre de construction. Par ailleurs, l'unité de projection 142 peut être configurée pour déterminer un premier axe intelligent principal défini à partir d'une bi-modalité marquée de la distribution des données initiale $B_0[1]$, et pour appliquer un seuil de découpe des modes de projection selon une valeur fixée, par exemple zéro dans le cas où les données sont centrées avant projection sur l'axe intelligent.

[0069] Ainsi, la génération d'un index effectuée par le dispositif de partitionnement de données 100 peut comprendre une ou plusieurs opérations de partitionnement associées chacune à une étape de détermination d'un axe intelligent spécifique et répétées plusieurs fois sur des distributions de données successives, de façon hiérarchique ou non. Par exemple, une étape de détermination d'un axe intelligent à partir d'une dis-

tribution des données $B_0[m]$ (avec ^{m} étant défini tel que $m > n$) peut être effectuée par l'unité de projection 142, telle que la distribution des données correspond à un des sous-ensembles de données générés à partir d'une étape antérieure de détermination d'un axe intelligent sur une autre distribution des données $B_0[n]$ du nœud parent $P[n]$.

[0070] Le module de traitement 140 (et/ou l'unité de projection 142) peut comprendre des unités de calcul indépendantes (non représentées sur les figures). Ainsi, le module de traitement 140 peut être configuré pour déterminer les différents axes intelligents de nœuds situés à une même profondeur i de l'arbre, de façon simultanée sur ces différentes unités de calcul. De façon générale, un nœud fils peut être construit à partir du moment où le nœud parent existe. De ce fait, l'unité de projection 142 peut être configurée pour générer un planificateur de tâches apte à administrer la construction de différentes nœuds donnés en fonction de l'existence d'un nœud parent et d'une ou plusieurs ressources de calcul disponibles par exemple.

[0071] Dans certains modes de réalisation, le module de traitement 140 peut comprendre une file de priorité comprenant au moins un élément. La génération d'un index effectuée par le module de traitement 140 peut alors être implémentée en utilisant cette file de priorité. Par exemple et de façon non limitative, un élément de la file de priorité peut correspondre au traitement d'une opération de partitionnement d'un nœud $P[n]$ (c'est-à-dire, à la détermination d'un axe intelligent à partir d'une distribution des données $B[n]$). En outre, le module de traitement 140 peut être configuré pour générer un ou plusieurs éléments dans la file de priorité en réponse au traitement d'un nœud $P[n]$. La file de priorité peut être de type FIFO (*First In, First Out* selon l'expression anglo-saxonne signifiant littéralement "premier entré, premier sorti ») et les éléments générés peuvent être ajoutés en queue de file de priorité.

[0072] En particulier, pour une structure de partitionnement binaire telle qu'illustrée sur la figure 3(a), une file de priorité de type FIFO engendre une construction de l'arbre en largeur d'abord (i.e. selon l'indice j). Ce type de parcours de construction l'arbre est particulièrement adapté à une distribution du calcul en générant des séquences contigües de tâches similaires correspondant aux traitements à effectuer sur les nœuds de l'arbre à un niveau de profondeur donnée. Les nœuds situés à une profondeur donnée correspondent à des tâches de partitionnement indépendantes pouvant donc être parallélisées sur des unités de calcul indépendantes.

[0073] Dans des modes de réalisation, le module de traitement 140 peut comprend une horloge interne associé à un instant ^{t}. A l'initialisation de l'index initial A_0, le paramètre ^{t} vérifie $t = t_0$ et à l'insertion d'un ensemble de données additionnel B_k, le paramètre ^{t} vérifie $t = t_k$. Par exemple et sans limitation, les instants t_0 et t_k peuvent

être des horodatages et/ou des intervalles de temps. Dans d'autres modes de réalisation, les instants t_0 et t_k peuvent être associés à des boucles d'itérations d'insertion de données.

[0074] Dans des modes de réalisation, le module de traitement 140 peut en outre être configuré pour sauvegarder un index (par exemple, l'index initial A_0 généré et/ou le ou les index A_k modifiés) dans le module de sauvegarde 160 du dispositif de partitionnement de données 100. En particulier, un index peut être sauvegardé en prenant compte des différents nœuds $P[n]$ (ou $P[i, j]$) qui le caractérisent. Par exemple, le module de sauvegarde 160 peut être configuré pour enregistrer chaque nœud $P[n]$ sous la forme d'une matrice d'éléments générées par le module de traitement 140.

[0075] Dans des modes de réalisation, à l'initialisation de l'index initial A_0, chaque nœud $P[n]$ (éventuellement noté $P[n](t_0)$) peut comprendre dans le module de sauvegarde 160 :

– l'indice de nœud ^{n} (ou équivalent à $[i, j]$) et éventuellement l'identité de l'index associé ;

– l'indice du nœud 'parent' (par exemple $^{\omega}$, ou pour un arbre binaire : $\left[(i-1), \left(\frac{j+1}{2}\right)\right]$ ou $\left[(i-1), \left(\frac{j}{2}\right)\right]$), et/ou l'indice du ou des nœuds fils (par exemple ^{m}, ou pour un arbre binaire : $[(i+1), (2j-1)]$ et $[(i+1), (2j)]$) ;

– un ensemble de données de construction $S_{build}[n]$ comprenant initialement la distribution de données $B_0[n]$ (ou $B_0[i, j]$) ;

– les paramètres (correspondant à des coordonnées pouvant être normalisées) de l'axe intelligent noté $V_0[n]$ (ou $V_0[i, j]$).

[0076] Selon certains modes de réalisation, chaque nœud $P[n]$ peut comprendre en outre, dans le module de sauvegarde 160, l'indice de nœud ^{n} (ou $[i, j]$) et optionnellement l'identité de l'index associé.

[0077] Avantageusement, un nœud $P[n]$ peut en outre comprendre, dans le module de sauvegarde 160, des paramètres de transformation de données notés $T[n]$. Une transformation de données peut être appliquée par le module de traitement 140 à chaque donnée du nœud $P[n]$ (c'est-à-dire, chaque donnée X d'un ensemble des données, comme par exemple $S_{build}[n]$). La transformation de données $T[n]$ peut être appliquée avant projection de la donnée sur un axe intelligent $V[n]$.

[0078] Dans des modes de réalisation, pour certains type d'axes intelligents (par exemple, pour un axe intelligent de type mini-kurtique), une transformation (ou procédé) de 'blanchiment' des données avant projection peut être appliqué. Un procédé de blanchiment correspond notamment à une succession d'étapes comprenant une étape

de centrage, une étape de décorrélation et une étape de normalisation suivant chacun des axes principaux. La transformation de type blanchiment des données $T[n](X)$ peut être définie par l'équation (02) suivante :

[0079] $$T[n](X) = D_{\sigma_n}^{-\frac{1}{2}} W_n^T (X - \mu_n) \quad (02)$$

[0080] Dans l'équation (02) :

– le terme μ_n fait référence à un vecteur de centrage de l'ensemble des données $S_{build}[n]$, c'est-à-dire par exemple la moyenne des données de constructions du nœud $P[n]$;

– le terme W_n fait référence à une matrice contenant un certain nombre de vecteurs propres normalisés qui définissent un sous-espace de projection des données ; et

– le terme $D_{\sigma_n}^{-\frac{1}{2}}$ fait référence à une matrice diagonale contenant l'inverse des valeurs propres associées à chaque vecteur propre.

[0081] Il est à noter que, la distribution de données $B_0[n]$ (ou $B_0[i, j]$) du nœud noté $P[n](t_0)$ est la distribution de données utilisée pour déterminer les coordonnées de l'axe intelligent $V[n]$. Ainsi à $t = t_0$, l'ensemble de données de construction $S_{build}[n]$ est égale à la distribution de données $B_0[n]$ issue de l'ensemble des données initial B_0.

[0082] Dans des modes de réalisation, chaque nœud $P[n]$ peut également comprendre, dans le module de sauvegarde 160, la représentation de la distribution des coordonnées de projection des données $B_0[n]$ sur l'axe intelligent $V_0[n]$.

[0083] L'homme du métier comprendra que chaque nœud feuille d'un index peut ne pas comprendre, dans le module de sauvegarde 160, d'indice de nœuds fils ni de paramètres d'axe intelligent. De façon similaire, le nœud $P[1]$ (ou $P[0,1]$) peut ne pas comprendre, dans le module de sauvegarde 160, d'indice de nœud parent.

[0084] Dans des modes de réalisation, à l'insertion d'un ensemble de données additionnel B_k, chaque nœud $P[n]$ (dans le module de sauvegarde 160) peut comprendre en outre l'ensemble de données insérées $S_{insert}[n]$ comprenant au moins une distribution de données $B_k[n]$, dite distribution de données insérée (ou ajoutée). Une distribution de données distribution insérée $B_k[n]$ est notamment issue de l'application de l'axe intelligent $V_0[\omega]$ du nœud parent $P[\omega]$ à un ensemble de données $B_k[\omega]$. L'axe intelligent $V_0[\omega]$ étant déterminé à l'initialisation de l'index initial A_0. Ainsi, à $t = t_1$, l'ensemble de données insérées $S_{insert}[n]$ peut être égal par exemple à l'ensemble de données inséré $B_1[n]$ au nœud $P[n]$.

[0085] Par ailleurs, dans des modes de réalisation, chaque nœud $P[n]$ peut comprendre, dans le module de sauvegarde 160, les représentations $H_{build}[n]$ et $H_{insert}[n]$

associées respectivement à l'ensemble de données de construction $S_{build}[n]$ et à l'ensemble de données insérées $S_{insert}[n]$ selon les mêmes coordonnées de l'axe intelligent $V_0[n]$.

[0086] Dans des modes de réalisation, lors d'une opération d'insertion d'une distribution de données dans un nœud $P[n]$, le module de traitement 140 peut être configuré pour appliquer une transformation de données $T_0[n]$ à la distribution de données insérée $B_k[n]$ avant projection sur l'axe intelligent $V_0[n]$.

[0087] En particulier, dans un cas d'utilisation d'un axe mini-kurtique, le module de traitement 140 peut être configuré pour mettre en œuvre le 'blanchiment' d'une donnée X compris dans une distribution de données insérée $B_k[n]$ en un nœud $P[n]$ selon la transformation de données $T[n](X)$ définie dans l'équation (02), puis pour exécuter le partitionnement de cette donnée d'insertion X selon l'équation (03) suivante :

[0088] $$V_n W_n(X - \mu_n) > 0 \quad (03)$$

[0089] L'équation (03) correspond à un embranchement dit conditionnel. Par exemple et sans limitation, la projection d'une donnée d'insertion X dans un index binaire permet d'aiguiller la donnée d'insertion X dans le nœud fils 'droit' si la condition (03) est vérifiée, ou dans le nœud fils 'gauche' sinon.

[0090] Tel qu'illustré sur la figure 4(a) et la figure 4(b), un ensemble de données additionnel B_k est inséré dans l'arbre de partitionnement binaire A_0 à un instant $t = t_k$. L'application de l'axe intelligent $V_0[0,1]$ du nœud $P[0,1]$ à la distribution de données $B_k[0,1]$ induit un partitionnement de données en deux distributions de données $B_k[1,1]$ et $B_k[1,2]$ qui sont injectées respectivement dans les nœuds $P[1,1]$ et $P[1,2]$. Dans ce cas, un ensemble de données de construction $S_{build}[n]$ est égal à $B_0[n]$, et un ensemble de données insérées $S_{insert}[n]$ est égal à $B_k[n]$.

[0091] Un index (binaire par exemple) peut être généré de sorte que, pour chaque groupement de données, il est possible de trouver un axe dont chaque mode ('gauche' ou 'droit' par exemple) est composé entièrement d'éléments de ce groupement. Alternativement, lors du partitionnement de données, un groupement de données peut être réparti sur plusieurs axes, en fonction de certains critères (par exemple un 'critère d'arrêt') fixés lors de la construction de l'arbre. Ainsi, si un histogramme 1D d'un index initial correspondant à la projection d'une distribution de données sur un axe intelligent est substantiellement modifié lors de l'insertion de nouvelles données (comme illustré sur la [Fig.4](a) et la [Fig.4](b)), alors ces nouvelles données correspondent à de nouveaux groupements absents des données initiales utilisées pour construire l'index initial. Ces nouvelles données sont donc non-stationnaires par rapport aux données de constructions initiales, ce qui implique une mise à jour du nœud et donc de

la définition de l'axe intelligent associé pour tenir compte des nouveaux groupements de données.

[0092] En particulier, les représentations $H_{build}[0,1]$ et $H_{insert}[0,1]$ illustrées sur la figure 4(b) montrent une distribution de données $B_k[1,1]$ différente par rapport à $B_0[1,1]$ (en particulier une distribution nulle ou quasi-nulle), ce qui reflète une non-stationnarité des données entre la distribution de données $B_0[0,1]$ et la distribution de données $B_k[0,1]$, chacune projetée sur l'axe intelligent $V_0[0,1]$ du nœud $P[0,1]$.

[0093] Comme illustré sur la figure 1, le module de traitement 140 peut comprendre une unité d'évaluation 144. Cette unité d'évaluation 144 peut être configurée pour comparer au niveau de chaque nœud $P[n]$ des distributions de données, et plus particulièrement pour comparer les représentations $H_{build}[n]$ et $H_{insert}[n]$ des ensembles de données $S_{build}[n]$ et $S_{insert}[n]$ projetées sur l'axe intelligent $V_0[n]$. En d'autres termes, l'unité d'évaluation 144 peut être configurée déterminer une valeur de comparaison notée $C[n]$ des ensembles de données $S_{build}[n]$ et $S_{insert}[n]$.

[0094] L'unité d'évaluation 144 peut en outre être configurée pour évaluer chacune des différentes valeurs de comparaisons $C[n]$ en fonction d'un seuil de non-stationnarité des données δ. Par exemple, l'unité d'évaluation 144 peut être apte à déterminer une valeur d'évaluation de la stationnarité de l'ensemble de données insérées $S_{insert}[n]$ par rapport à l'ensemble de données de construction $S_{build}[n]$. Ainsi, en chaque nœud $P[n]$, la valeur d'évaluation est notamment issue de la valeur de comparaison $C[n]$ et est examinée en fonction du seuil de non-stationnarité des données δ, afin de détecter chaque non-stationnarité de données insérées dans un index.

[0095] Dans des modes de réalisation, le dispositif de partitionnement de données 100 peut être configuré pour effectuer, au niveau du nœud $P[n]$, une ou plusieurs opérations d'insertion de données $\{B_k[n]\}_{k\in[1,..,K]}$, le nœud $P[n]$ pouvant alors comprendre également, dans le module de sauvegarde 160 :

– l'ensemble de données de construction $S_{build}[n] = B_0[n]$; et

– l'ensemble de données insérées $S_{insert}[n] = \{B_k[n]\}_{k\in[1,..,K]}$.

[0096] Selon certains modes de réalisation, la détermination des valeurs de comparaison $C[n]$ de données par l'unité d'évaluation 144 peut être effectuée en utilisant toute méthode d'évaluation de dissemblance entre deux distributions de données adaptée, comme par exemple, une méthode de calcul de distance entre représentations (ou histogrammes) $H_{build}[n]$ et $H_{insert}[n]$.

[0097] Un exemple de méthode d'évaluation de dissemblance entre deux distributions qui peut être utilisée est la méthode de calcul d'une distance appelée « distance du

cantonnier» ou encore EMD (acronyme de l'expression anglo-saxonne « earth mover's distance »), décrite dans l'article "A distance metric for multidimensional histograms" de M.Werman, S. Peleg, and A. Rosenfeld 1985, Computer Vision, Graphics, and Image Processing, 32(3):328–336.

[0098] En particulier, la mise en œuvre de la méthode de calcul EMD en un nœud $P[n]$ permet de déterminer la valeur de comparaison $C[n]$ de données selon l'équation (04) suivante :

[0099] $$C[n] = EMD(H_{build}[n],\ H_{insert}[n]) \quad (04)$$

[0100] Dans un premier mode de réalisation, la valeur d'évaluation de la stationnarité peut être la valeur de comparaison $C[n]$ déterminée en un nœud $P[n]$ selon l'équation (04). Par ailleurs, le seuil de non-stationnarité δ permettant de détecter une non-stationnarité entre l'ensemble de données de construction $S_{build}[n]$ et l'ensemble de données insérées $S_{insert}[n]$ peut être prédéfini ou prédéterminé. Dans ce cas, l'unité d'évaluation 144 peut être par exemple configurée pour vérifier si la valeur de comparaison $C[n]$ est strictement supérieure au seuil de non-stationnarité δ, tel que :

[0101] $$C[n] > \delta \quad (05)$$

[0102] Dans le mode de réalisation où la condition de non-stationnarité des données est définie par l'équation (05), si la valeur de la comparaison $C[n]$ est strictement supérieure au seuil de non-stationnarité δ, alors une non-stationnarité au nœud $P[n]$ est détectée.

[0103] Dans d'autres modes de réalisation, le seuil de non-stationnarité δ peut être ajusté par rapport à une probabilité de fausse alarme fixée notée ζ (qui peut être par exemple une probabilité de fausse alarme souhaitée ou cible), définie à l'échelle de l'index considéré. Dans ce cas, le seuil de non-stationnarité $\delta(\zeta)$ peut être déterminé en fonction du nombre de nœuds N de l'index. Pour une probabilité de fausse alarme ζ constante, le module de traitement 140 détectera un nombre constant de non-stationnarités (égal à $\zeta \times N$), à chaque itération d'insertion de données en flux.

[0104] L'unité d'évaluation 144 peut être alors configurée pour mettre en œuvre un test statistique basé sur un principe de détection a contrario de non-stationnarités. Un tel principe est par exemple décrit dans l'article "From gestalt theory to image analysis: a probabilistic approach" de A. Desolneux, L. Moisan, and J.-M. Morel, 2007, Springer Science & Business Media, volume 34.

[0105] En particulier, l'unité d'évaluation 144 peut être configurée pour déterminer à l'échelle d'un index considéré :

– une première hypothèse telle qu'une « hypothèse nulle notée H_0 » correspond à l'hypothèse selon laquelle il n'y a pas de non-stationnarité détectée au nœud $P[n]$; et

– une variable aléatoire C associée aux observations $\{C[n]\}_{n=1,\ldots N}$ (c'est-à-dire aux différentes valeurs de comparaisons à chaque nœud $P[n]$, une valeur de comparaison correspondant à une distance entre les histogrammes $H_{build}[n]$ et $H_{insert}[n]$).

[0106] L'unité d'évaluation 144 peut en outre être apte à déterminer à chaque nœud $P[n]$ une distribution cumulative notée $F_E(C[n])$ (aussi appelée fonction de répartition) telle que définie dans l'équation (06) suivante :

[0107] $$F_E(C[n]) = P(C > \delta | H0) \quad (06)$$

[0108] Dans ces modes de réalisation, la détermination de la valeur d'évaluation de la stationnarité en un nœud $P[n]$ consiste à évaluer la distribution cumulative $F_E(C[n])$ selon l'équation (06).

[0109] Dans ce cas, l'unité d'évaluation 144 peut être apte à comparer la valeur d'évaluation (ou valeur de la fonction de répartition $F_E(C[n])$) par rapport à la probabilité de fausse alarme fixée ζ et indépendante du nœud $P[n]$ considéré, tel que :

[0110] $$F_E(C[n]) < \zeta \quad (07)$$

[0111] Dans le mode de réalisation correspondant à l'équation (07), si au nœud $P[n]$, la valeur de la fonction de répartition $F_E(C[n])$ est supérieure au taux de fausses alarmes fixé ζ, alors une non-stationnarité des données insérées est détectée au nœud $P[n]$. Inversement, si la valeur de la fonction de répartition $F_E(C[n])$ est strictement inférieure au taux de fausses alarmes fixé ζ, alors l'absence de non-stationnarité des données insérées est vérifiée au nœud $P[n]$. En d'autres termes, la vérification de cette condition signifie que la probabilité d'observer des valeurs supérieures ou égales à $C[n]$ est trop faible pour que la valeur $C[n]$ ait été observée par hasard.

[0112] Ainsi, si au nœud $P[n]$ la valeur de la fonction de répartition $F_E(C[n])$ est inférieure à un taux de fausses alarmes prédéfini ζ, une seconde hypothèse notée H_1 correspondant à la « détection d'une non-stationnarité » pour la valeur $C[n]$, l'hypothèse H_0 n'est vraisemblablement pas valide pour expliquer cette réalisation de la variable aléatoire C. Autrement dit, la valeur $C[n]$ contredit la loi de C sous l'hypothèse nulle H_0.

[0113] Il est à noter que l'utilisation du principe de détection « a contrario » de non-stationnarités permet au seuil de non-stationnarité δ de s'ajuster en fonction des opérations d'insertion et/ou des opérations de partitionnement par opposition à une valeur du seuil de non-stationnarité δ préalablement établie.

[0114] Dans le cas particulier illustré sur la figure 4(b), les nouvelles données B_k correspondent exclusivement à un mode préalablement existant puisque les repré-

sentations $H_{build}[0,1]$ et $H_{insert}[0,1]$ montrent un sous-ensemble de données $B_k[1,1]$ vide par rapport à $B_0[1,1]$, chaque mode associé à une projection de données sur un axe intelligent comprenant un poids. Ainsi, dans ce cas, l'exclusivité de répartition des données selon une distribution de données $B_k[1,2]$ modifie uniquement le poids des modes, et non les paramètres de position et de forme des groupements de données des nœuds fils. Dans le mode de réalisation d'utilisation d'axes mini-kurtiques pour le partitionnement de donnée, il n'est pas nécessaire de re-calculer l'axe intelligent $V_0[n]$ puisque l'axe défini initialement reste optimal au sens du minimum de kurtosis, même en cas de changement des poids des groupements. Dans ce cas particulier et selon un mode de réalisation simple avec un seuil de non-stationnarité préalablement établi δ, la distance EMD entre les représentations $H_{build}[0,1]$ et $H_{insert}[0,1]$ définie à l'équation (05) est inférieure au seuil de non-stationnarité δ et donc une non-stationnarité n'est pas détectée. Alternativement, selon un mode de réalisation complexe avec un seuil de non-stationnarité ajusté $\delta(\zeta)$ par rapport à une probabilité de fausse alarme ζ, une non-stationnarité peut être détectée.

[0115] Il est à noter que, la reconstruction de la structure par mises à jour successives n'a d'impact que sur la quantité de calculs effectués, et non sur la qualité de l'index (en termes de pureté moyenne des feuilles). Ainsi, dans certains modes de réalisation, le dispositif de partitionnement de données 100 peut être configuré pour exécuter (ou déclencher) au nœud $P[n]$ une mise à jour dite 'mise à jour induite par comparaison'. Cette mise à jour induite par comparaison est exécutée au nœud $P[n]$ en réponse à la détection d'une non-stationnarités de données au nœud $P[n]$ suite à une opération d'insertion de données $B_k[n]$. Dans ce cas, le dispositif de partitionnement de données 100, via l'unité de projection 142 par exemple, peut alors être configuré pour effectuer une opération de partitionnement sur l'ensemble de données de construction alors modifié et défini par $S_{build}[n] = \{B_0[n], B_k[n]\}$. Cette opération de partitionnement génère ainsi de nouvelles coordonnées d'axe intelligent, noté $V_k[n]$, alors enregistrées dans le module de sauvegarde 160. L'ensemble de données insérées $S_{insert}[n]$ du nœud $P[n]$ (dans le module de sauvegarde 160) est alors vidé. La représentation $H_{build}[n]$ associée à l'ensemble de données de construction $S_{build}[n]$ selon les coordonnées de l'axe intelligent $V_k[n]$ peut alors être mis à jour par le dispositif de partitionnement de données 100 (dans le module de sauvegarde 160), la représentation $H_{insert}[n]$ étant alors nulle.

[0116] Par exemple, tel que illustré sur la figure 4(a), un ensemble de données additionnel B_k est inséré dans l'arbre de partitionnement binaire A_0 à un instant $t = t_k$. Suite à la détection d'une non-stationnarité de données au nœud $P[0,1]$ entre la distribution de

données $B_0[0,1]$ et la distribution de données $B_k[0,1]$ projetées sur l'axe intelligent $V_0[0,1]$ et illustrées sur la figure 4(b), à un instant $t = t_{k+}$, l'arbre de partitionnement binaire est mis à jour, et un axe intelligent $V_k[0,1]$ est généré à partir d'une opération de partitionnement sur l'ensemble de données de construction modifié correspondant alors au nouvel ensemble de données de construction $S_{build}[n] = \{B_0[n], B_k[n]\}$, également noté $S_{build}[n](t_{k+})$. En particulier, $S_{build}[n](t_{k+}) = S_{build}[n](t_k) \cup S_{insert}[n](t_k)$ et $S_{insert}[n](t_{k+})$ est vidé. La projection de $S_{build}[n]$ sur l'axe intelligent $V_k[0,1]$ (i.e. la représentation $H_{build}[0,1]$ illustrée sur la figure 5(b)) induit un partitionnement de données en deux distributions de données, notées $\{B_0, B_k[1,1]\}$ et $\{B_0, B_k[1,2]\}$, injectées respectivement dans les nœuds $P[1,1]$ et $P[1,2]$ comme représenté sur la [Fig.5](a).

[0117] Dans des modes de réalisation, le dispositif de partitionnement de données 100 peut être configuré pour exécuter, au nœud $P[n]$, une autre mise à jour dite 'mise à jour résultante'. Cette mise à jour résultante est exécutée au nœud $P[n]$ en réponse à une mise à jour exécutée au nœud parent $P[\omega]$. Par exemple, une mise à jour résultante peut être exécutée au niveau du nœud $P[n]$, en réponse à une mise à jour induite par comparaison exécutée au nœud parent $P[\omega]$ dont les coordonnées d'axe intelligent $V_k[\omega]$ ont été modifiés, ainsi que les sous-ensembles de données résultant de la projection de l'ensemble de données de construction $S_{build}[\omega]$ sur l'axe intelligent $V_k[\omega]$. L'ensemble de données insérées $S_{insert}[n]$ du nœud $P[n]$ est alors vidé, dans le module de sauvegarde 160. La représentation $H_{build}[n]$ associée à l'ensemble de données de construction $S_{build}[n]$ selon les coordonnées de l'axe intelligent $V_k[n]$ peut alors être mis à jour par le dispositif de partitionnement de données 100 dans le module de sauvegarde 160, la représentation $H_{insert}[n]$ étant alors nulle.

[0118] Dans l'exemple représenté sur la figure 5(a) et la figure 5(b), les deux distributions de données $B_0, B_k[1,1]$ et $B_0, B_k[1,2]$ sont insérées respectivement dans les nœuds $P[1,1]$ et $P[1,2]$ suite à la mise à jour induite par comparaison exécutée au nœud parent $P[0,1]$. En réponse à cette mise à jour au nœud $[0,1]$, des mises à jour résultantes peuvent être exécutées aux nœuds $P[1,1]$ et $P[1,2]$ telles que des axes intelligent $V_k[1,1]$ et $V_k[1,2]$ peuvent être générés. De façon similaire, des mises à jour résultantes peuvent être exécutées aux nœuds $P[2,1]$, $P[2,2]$, $P[2,3]$ et $P[2,4]$ en réponse aux mises à jour résultantes précédentes exécutées aux nœuds $P[1,1]$ et $P[1,2]$.

[0119] Il est à noter que lorsqu'une non-stationnarité est détectée en un nœud $P[n]$,

l'ensemble de la sous-structure de l'index est mis à jour en dessous de ce nœud. Ainsi, dans des modes de réalisation, le dispositif de partitionnement de données 100 peut être apte à répertorier, en chaque nœud $P[n]$, la ou les mises à jour à exécuter. Par exemple et sans limitation, suite à une opération d'insertion de données, un nœud $P[n]$ peut être associé à une mise à jour induite par comparaison et/ou une ou plusieurs mises à jour résultantes d'une mise à jour induite par comparaison à effectuer en un nœud parent. Dans ce cas, le dispositif de partitionnement de données 100 peut être configuré pour exécuter en priorité (au moyen de l'opération consistant à répertorier des mises à jour à exécuter) la mise à jour induite par comparaison associée au nœud parent le plus proche de la racine de l'index (c'est-à-dire ayant la profondeur i par exemple la plus faible dans un arbre binaire), puis pour exécuter les mises à jour résultante associées à l'ensemble des nœuds fils, par exemple selon une profondeur i croissante dans l'arbre par exemple.

[0120] Avantageusement, dans le mode de réalisation associé à l'équation (07) et utilisant un seuil de non-stationnarité ajusté $\delta(\zeta)$ par rapport à une probabilité de fausse alarme ζ, le dispositif de partitionnement de données 100 peut être configuré pour conserver les valeurs historiques $S_{build}[n]$ issues des itérations précédentes, selon un facteur d'oubli noté ξ, par exemple dans le module de sauvegarde 160.

[0121] L'utilisation du facteur d'oubli ξ permet notamment de contrer l'effet de changement d'échelle des valeurs $C[n]$ entre itérations k de flux de données successives à insérer, notamment lors de la détermination de la fonction de répartition F_E en chaque nœud $P[n]$. Ainsi, les mises à jour induites par comparaison dépendent de l'historique des données pour contrer les changements de dynamiques des valeurs $C[n]$ entre itérations de flux.

[0122] En effet, il est à noter qu'un index initial A_0 est construit sur une 'vision' très partielle d'un ensemble global de données $\{B_0[n], B_1[n], B_K[n]\}$. Ainsi, à une première insertion de données additionnelles $B_1[n]$, le dispositif de partitionnement de données 100 peut détecter un grand nombre de non-stationnarités, avec des valeurs élevées pour chacune des valeurs de comparaison $C[n]$. Dans le mode de réalisation simple avec un seuil de non-stationnarité préalablement déterminé δ, à une $k^{ième}$ insertion de données additionnelles $B_k[n]$, le dispositif de partitionnement de données 100 peut déterminer des valeurs plus faibles pour chacune des comparaisons $C[n]$ et ainsi détecter moins, voire beaucoup moins de non-stationnarités, ce qui engendre notamment une décroissance du nombre de mises à jour induites par comparaison de la structure d'index avec le nombre d'insertions. Alternativement, selon le mode de réalisation de l'équation (07) utilisant un seuil de non-stationnarité ajusté $\delta(\zeta)$ par rapport

à une probabilité de fausse alarme ζ, le nombre de mises à jour induites par comparaison de la structure d'index reste constant. Ainsi, un facteur d'oubli ξ entre itérations permet dans ce cas de « régler » la sensibilité du mécanisme de détection des non-stationnarités de données insérées, et permet d'autoriser la mise à jour de l'index dans les itérations ultérieures de flux, même pour des valeurs plus faibles de valeurs de comparaison $C[n]$.

[0123] Il est à noter que les enregistrements en chaque nœud $P[n]$ de l'indice du nœud parent et/ou de l'indice du ou des nœuds fils, de l'ensemble de données de construction, des paramètres de l'axe intelligent et de l'ensemble de données insérées sont suffisants pour effectuer des insertions de nouvelles données dans un index et pour effectuer des améliorations complémentaires des modes de réalisation de l'invention. En particulier, l'enregistrement spécifique des indices d'insertion des données, à chaque itération d'opération d'insertion et dans chaque nœud, permet une indexation multi-résolution du jeu de données complet $\{B_0, B_k\}$.

[0124] Le procédé algorithmique de découpage récursif de l'espace, selon les modes de réalisation de l'invention, permet la distribution des calculs sur des unités de traitement parallèles avec mémoires non partagées, ce qui rend possible l'analyse de grandes quantités de données.

[0125] Comme illustré sur la [Fig.1], le système 10 peut notamment comprendre en outre un module d'exploitation 18 configuré pour appliquer des traitements de données spécifiques à l'aide d'algorithmes, par exemple fortement supervisés, et selon des conditions d'application strictes sur des données stationnaires et des volumes de données restreints.

[0126] Le module d'exploitation 18 peut également être configuré pour évaluer la performance de la génération d'index (i.e. la génération d'arbre), c'est-à-dire pour effectuer une évaluation 'sémantique' de la cohérence des partitions de la structure, et en particulier pour calculer la performance en termes de pureté moyenne des partitions formées.

[0127] Dans des modes de réalisation, les données de constructions et données insérées en flux peuvent être étiquetées selon une pluralité de L étiquettes. Ces étiquettes peuvent par exemple être prédéfinies et au moins certaines des données (ou éléments) des sources de données 12 peuvent être associées (ou annotées) à une ou plusieurs de ces étiquettes.

[0128] Le module d'exploitation 18 peut ainsi être configuré pour calculer un score de pureté d'une ou plusieurs partitions de données formées dans un index binaire, par exemple aux nœuds $P[i, j]$. Le module d'exploitation 18 peut également être configuré pour déterminer un score de pureté moyenne d'une même profondeur i d'un

arbre binaire, en particulier à partir de scores calculés de pureté des partitions de données.

[0129] Il est à noter qu'une partition de données peut être associée à un score de pureté égal à 1, si elle est constituée entièrement d'éléments associés à une même étiquette. Inversement, une partition de données peut être associée à un score de pureté égale à $1/\mathrm{L}$, si elle est constituée d'éléments associés de manière uniforme à toute la pluralité de L étiquettes. Le calcul de score de pureté d'une partition de données au nœud $P[i, j]$ peut être défini comme le rapport entre le cardinal de l'étiquette majoritaire des éléments de la partition de données et le nombre total de données annotées au nœud $P[i, j]$.

[0130] Avantageusement, le score de pureté moyenne à une même profondeur i d'un l'arbre binaire peut correspondre à une moyenne pondérée des scores de pureté calculés de l'ensemble des nœuds $P[i, j]$, i étant prédéfini. La pondération peut être choisie comme un nombre de données annotées par partition. Ce choix de pondération a l'avantage d'éviter aux étiquettes ayant peu d'éléments annotés de peser de façon excessive dans la détermination de la pureté moyenne.

[0131] A une même profondeur i d'un l'arbre binaire, un score de pureté moyenne peut être égal à 1, si chaque partition de données des nœuds $P[i, j]$ associés est pure, c'est-à-dire constituée uniquement d'éléments associés à une même étiquette.

[0132] Avantageusement, le module d'exploitation 18 peut également être configuré pour déterminer (ou évaluer) la qualité d'un index de partitionnement de données à partir d'un ou plusieurs scores de pureté moyenne déterminés.

[0133] Les différents éléments du système 10, c'est-à-dire notamment le dispositif de partitionnement de données 100, les sources de données 12 et le module d'exploitation 18, peuvent être couplés à un bus de données 11.

[0134] La [Fig.6] est un organigramme décrivant un procédé de construction d'un arbre initial de partitionnement de données, mis en œuvre par un dispositif de partitionnement de données 100, selon des modes de réalisation de l'invention. Ce procédé est notamment agnostique par rapport au type choisi d'axe intelligent à calculer.

[0135] A l'étape 600, un ensemble de donnée initial B_0 est reçu, et le premier nœud de l'index est initialisé, tel que pour $n = 1$, l'ensemble de données de construction du nœud $P[1]$ est défini par $S_{build}[1] = B_0$, ou pour une structure binaire $S_{build}[0,1] = B_0$.

[0136] A l'étape 602, différents paramètres de constructions de l'index initial A_0 sont reçus. Ces paramètres de constructions peuvent comprendre par exemple la profondeur de l'arbre de partitionnement binaire L, et/ou le ou les types d'axes intelligents choisis, et des éléments d'utilisation associés.

[0137] A l'étape suivante ou aux étapes 604 et 608, le ou les indices du premier nœud sont initialisés. Par exemple, ^{n} est initialisé à $n = 1$, ou pour une structure binaire les indices i et j sont initialisés respectivement à $i = 1$ et $j = d$, d désignant un paramètre d'incrémentation de l'indice j, et étant lui-même initialisé à la valeur $d = 1$, à l'étape 606.

[0138] A l'étape 610, l'axe intelligent $V_0[i, j]$ associé à la distribution des données $B_0[i, j]$ (ou plus généralement l'ensemble de données de construction $S_{build}[i, j]$) du nœud $P[i, j]$ est déterminé.

[0139] A l'étape 612, la distribution des données $B_0[i, j]$ (ou plus généralement l'ensemble de données de construction $S_{build}[i, j]$) est projetée sur l'axe intelligent $V_0[i, j]$ résultant de la détermination des distributions de données issues du partitionnement de la distribution du nœud $P[i, j]$. Les nœuds fils du nœud $P[i, j]$ sont alors initialisés, par exemple aux valeurs respectives $P[i+1, d]$ et $P[i+1, d+1]$ pour une structure binaire.

[0140] Aux étapes 614 et 616, si les indices i et j atteignent les valeurs de certains paramètres de constructions de l'index, alors la construction de l'arbre de partitionnement A_0 est terminée, et en particulier l'arbre est enregistrée à l'étape 618 via notamment la sauvegarde des différentes éléments caractéristiques liées aux nœuds $P[n]$. Sinon, le ou les indices du nœud sont incrémentés, tel que par exemple $n = n + 1$, ou pour une structure binaire $i = i + 1$ (à l'étape 620) et $d = d + 2$ (à l'étape 622).

[0141] L'homme du métier comprendra aisément que certaines étapes peuvent être réalisées de manière simultanée et/ou selon un ordre différent, par exemple selon un ordre défini par le module de traitement 140.

[0142] La [Fig.7] est un organigramme décrivant un procédé d'insertion de données dans un arbre de partitionnement, ainsi que d'évaluation des données insérées par rapport aux données de construction de cet arbre, mis en œuvre par un dispositif de partitionnement de données 100, selon des modes de réalisation de l'invention.

[0143] A l'étape 700, l'arbre de partitionnement A_0 (via les différentes éléments liées aux nœuds $P[n]$) est reçu. En outre, à l'étape 700, un ensemble de donnée additionnel B_k est reçu, et le premier nœud de l'index est initialisé tel qu'à $n = 1$, l'ensemble de donnée additionnel B_k est ajouté à l'ensemble de données insérées $S_{insert}[1]$ du nœud $P[1]$, ou pour une structure binaire $S_{insert}[0,1]$.

[0144] Les étapes 704, 706, 708, 714, 716, 720 et 722 sont des étapes d'initialisation et d'incrémentation du ou des indices associées aux nœuds de l'index de partitionnement. L'homme du métier comprendra aisément que ces étapes sont analogues aux étapes 604, 606, 608, 614, 616, 620 et 622 associées aux indices des nœuds dans le procédé

de construction d'un arbre de partitionnement de données.

[0145] A l'étape 710, l'ensemble de données insérées $S_{insert}[i, j]$ (et notamment la distribution des données $B_k[i, j]$) est projetée sur l'axe intelligent $V_0[i, j]$, résultant de la détermination des distributions de données insérées $B_k[i+1, d]$ et $B_k[i+1, d+1]$ issues du partitionnement de la distribution des données $B_k[i, j]$.

[0146] A l'étape 712, la projection de l'ensemble de données insérées $S_{insert}[i, j]$ (comprenant notamment la distribution des données $B_k[i, j]$) est comparée à la projection de l'ensemble de données de construction $S_{build}[i, j]$ (comprenant notamment la distribution des données $B_0[i, j]$) afin de déterminer une valeur de comparaison $C[i, j]$.

[0147] A l'étape 718, la modification de l'arbre de partitionnement A_0 est terminée et les distributions de données insérées et leur projection sur les axes intelligents sont sauvegardées pour chaque nœud $P[n]$.

[0148] L'homme du métier comprendra aisément que le procédé d'insertion de données dans un arbre de partitionnement et le procédé d'évaluation des données insérées peuvent être réalisés de manière simultanée.

[0149] La [Fig.8] est un organigramme décrivant un procédé de mise à jour d'un arbre de partitionnement, mis en œuvre par un dispositif de partitionnement de données 100, selon des modes de réalisation de l'invention.

[0150] A l'étape 800, l'arbre de partitionnement A_0 modifié par l'insertion de données, ainsi que les informations d'évaluation (déterminées à l'étape 712 par exemple) sont reçus.

[0151] A l'étape 802, le type de seuil de non-stationnarité est déterminé, comme par exemple le seuil préalablement déterminé δ ou à ajuster $\delta(\zeta)$ par rapport à une probabilité de fausse alarme ζ à déterminer. En fonction de ces seuils, des éléments d'évaluation de la stationnarité des données peuvent être déterminées, comme par exemple l'hypothèse nulle et la valeur aléatoire pour la 'détection a contrario'.

[0152] A l'étape 804, les valeurs d'évaluation de la stationnarité sont déterminées en chaque nœud $P[n]$ en fonction des valeurs de comparaison $C[n]$, puis sont comparées au seuil de non-stationnarité afin de déterminer si l'ensemble de données insérées $S_{insert}[n]$ (comprenant notamment la distribution des données $B_k[n]$) sont stationnaires par rapport à l'ensemble de données de construction $S_{build}[n]$ (et notamment la distribution des données $B_0[n]$). Par exemple, une valeur d'évaluation de la stationnarité des données est égale la valeur de comparaison $C[i, j]$ ou déterminée en fonction de la valeur de comparaison $C[i, j]$ pour obtenir une fonction de répartition $F_E(C[n])$ pour la 'détection a contrario'.

[0153] A l'étape 806, la priorisation des mises à jours induites par comparaison est dé-

terminée à partir de l'ensemble des non-stationnarités détectées pour obtenir une liste d'un certain nombre Q de nœuds $P[n_q]$ à mettre à jour.

[0154] Les étapes 808, 816 et 820 sont des étapes d'initialisation et d'incrémentation du ou des indices associées à la liste de nœuds $P[n_q]$, et de l'index de partitionnement à mettre à jour (mises à jours induites par comparaison).

[0155] A l'étape 810, au nœud $P[n_q]$, l'ensemble de données inséré $S_{insert}[n_q]$ est vidé et l'ensemble de données de construction $S_{build}[n_q]$ est réinitialisé (i.e. modifié) pour comprendre toutes les distributions des données $\{B_0[n_q], \ldots B_k[n_q]\}$.

[0156] A l'étape 812, l'axe intelligent $V_k[n_q]$ associé à l'ensemble de données de construction $S_{build}[n_q]$ est déterminé (tel que pour l'étape 610). Par ailleurs, l'ensemble de données de construction $S_{build}[n_q]$ est projetée sur l'axe intelligent $V_k[n_q]$ résultant de la détermination des nouveaux ensembles de données de construction des nœuds fils (tel que pour l'étape 612).

[0157] A l'étape 814, pour chaque nœud fils modifié issue de la mise à jour du nœud $P[n_q]$, une mise à jour résultante est effectuée en reprenant par exemple les étapes 610 et 612 du procédé de construction de l'arbre.

[0158] A l'étape 818, l'arbre de partitionnement mis à jour A_k est terminé et les ensemble de données résultantes (et leur projection sur les axes intelligents) sont sauvegardés pour chaque nœud $P[n]$.

[0159] L'invention peut être mise en œuvre dans de nombreuses applications et dans des domaines techniques variés. Le dispositif de partitionnement de données 100 et les procédés selon les modes de réalisation de l'invention peuvent être utilisés par exemple dans des systèmes industriels (systèmes de cybersécurité, processus de fabrication par exemple) ou militaire, dans le domaine bancaire ou fiscal, dans des dispositifs d'imagerie (imagerie satellitaire par exemple), de diagnostic médical ou d'aide au diagnostic médical. L'invention s'applique plus généralement à tout domaine pour lequel il existe un besoin de partitionnement de données.

[0160] Par exemple et sans limitations, les dispositifs de partitionnement de données 100 peuvent être utilisés sur des images de surveillance satellitaire et aérienne haute ré-solution, notamment pour la détection d'anomalies, la détection d'objets, la détection de changements, la visualisation, ou encore la recherche de cibles.

[0161] L'homme du métier comprendra que l'invention peut être mise en œuvre en tant que programme d'ordinateur comportant des instructions pour son exécution. Le programme d'ordinateur peut être enregistré sur un support d'enregistrement lisible par un processeur. La référence à un programme d'ordinateur qui, lorsqu'il est exécuté, effectue l'une quelconque des fonctions décrites précédemment, ne se limite pas à un

programme d'application s'exécutant sur un ordinateur hôte unique. Au contraire, les termes programme d'ordinateur et logiciel sont utilisés ici dans un sens général pour faire référence à tout type de code informatique (par exemple, un logiciel d'application, un micro logiciel, un microcode, ou toute autre forme d'instruction d'ordinateur) qui peut être utilisé pour programmer un ou plusieurs processeurs pour mettre en œuvre des aspects des techniques décrites ici. Les moyens ou ressources informatiques peuvent notamment être distribués ("Cloud computing"), éventuellement selon des technologies de pair-à-pair. Le code logiciel peut être exécuté sur n'importe quel processeur approprié (par exemple, un microprocesseur) ou cœur de processeur ou un ensemble de processeurs, qu'ils soient prévus dans un dispositif de calcul unique ou répartis entre plusieurs dispositifs de calcul (par exemple tels qu'éventuellement accessibles dans l'environnement du dispositif). Le code exécutable de chaque programme permettant au dispositif programmable de mettre en œuvre les processus selon l'invention, peut être stocké, par exemple, dans le disque dur ou en mémoire morte. De manière générale, le ou les programmes pourront être chargés dans un des moyens de stockage du dispositif avant d'être exécutés. L'unité centrale peut commander et diriger l'exécution des instructions ou portions de code logiciel du ou des programmes selon l'invention, instructions qui sont stockées dans le disque dur ou dans la mémoire morte ou bien dans les autres éléments de stockage précités.

[0162] Le dispositif de partitionnement de données 100 peut être implémenté sur une unité de calcul distribué comprenant une pluralité de cœurs physiques (par exemple 16 cœurs). Les modes de réalisation de l'invention sont particulièrement adaptés à une telle implémentation distribuée ainsi qu'à un portage sur des technologies hardware multi-cœur, ce qui permet d'obtenir une réponse rapide à une requête utilisateur quelle que soit la taille du modèle de classification considéré. Le dispositif de partitionnement de données 100 peut être est implémenté dans différents langages, tels que le langage C++.

[0163] L'invention n'est pas limitée aux modes de réalisation décrits ci-avant à titre d'exemple non limitatif. Elle englobe toutes les variantes de réalisation qui pourront être envisagées par l'homme du métier. En particulier, l'homme du métier comprendra que l'invention n'est pas limitée aux différents procédés de comparaison des données et d'évaluation de leur stationnarité.

Revendications

[Revendication 1] Dispositif de partitionnement de données (100) apte à recevoir des données transmises par au moins une source de données (12), les données comprenant un ensemble de données initial B_0 et un ou plusieurs ensembles de données additionnels B_k, le dispositif (100) étant configuré pour effectuer :
- au moins une opération de partitionnement initiale pour générer un arbre de partitionnement de données initial A_0 à partir dudit ensemble des données initial B_0, ledit arbre étant associé à N sous-structures $P[n]$ comprenant chacune une distribution de données de construction $B_0[n]$ issue de l'ensemble de données initial B_0, et
- au moins une opération d'insertion pour insérer ledit ensemble de données additionnels B_k dans l'arbre de partitionnement de données initial A_0, lesdites sous-structures $P[n]$ comprenant en outre au moins une distribution de données insérée $B_k[n]$ issue de l'ensemble de données additionnels B_k,
le dispositif (100) étant en outre configuré pour effectuer au niveau de chaque sous-structure $P[n]$ une comparaison $C[n]$ de ladite distribution de données de construction $B_0[n]$ avec ladite distribution de données insérée $B_k[n]$ et pour déterminer une valeur d'évaluation de la stationnarité des données insérées par rapport aux données de construction de l'arbre de partitionnement de données initial A_0 à partir de ladite comparaison,
le dispositif de partitionnement de données (100) étant configuré pour mettre à jour le partitionnement de données dudit arbre à partir de ladite valeur d'évaluation.

[Revendication 2] Dispositif (100), selon la revendication 1, dans lequel ladite opération de partitionnement initiale comprend la détermination d'un axe de projection $V_0[n]$ de ladite distribution de données de construction $B_0[n]$ pour former au moins deux sous-ensembles de données distincts issus de la distribution de données de construction $B_0[n]$, et dans lequel ladite opération d'insertion dudit ensemble de données additionnels B_k est effectué au niveau de chaque sous-structure $P[n]$ en prenant en compte ledit axe de projection $V_0[n]$.

[Revendication 3] Dispositif (100), selon la revendication 2, dans lequel ladite comparaison $C[n]$ comprend une détermination d'une distance entre la

projection de ladite distribution de données de construction $B_0[n]$ et la projection de ladite au moins une distribution de données insérée $B_k[n]$, ladite distance étant par exemple la distance EMD.

[Revendication 4] Dispositif (100), selon l'une des revendication précédentes, dans lequel le dispositif (100) est en outre configuré pour effectuer au moins une nouvelle opération de partitionnement, en réponse à une détection d'une condition de non-stationnarité des données, ladite nouvelle opération de partitionnement de données étant effectuée au niveau d'au moins une sous-structure $P[n]$ à partir de ladite distribution de données $B_0[n]$ et de ladite au moins une distribution de données insérée $B_k[n]$ de manière à générer un arbre de partitionnement de données modifié A_k, ladite condition de non-stationnarité des données étant vérifiée si ladite valeur d'évaluation de la stationnarité des données est supérieure à un seuil de non-stationnarité des données δ.

[Revendication 5] Dispositif (100), selon la revendication 4, et dans lequel ladite nouvelle opération de partitionnement comprend la détermination d'un nouvel axe de projection $V_k[n]$ d'un ensemble de données de construction $S_{build}[n]$ pour former au moins deux nouveaux sous-ensembles de données distincts issus dudit ensemble de données de construction $S_{build}[n]$, ledit ensemble de données de construction $S_{build}[n]$ comprenant ladite distribution de données de construction $B_0[n]$ et ladite au moins une distribution de données insérée $B_k[n]$.

[Revendication 6] Dispositif (100), selon l'une des revendications 4 ou 5, dans lequel ledit seuil de non-stationnarité des données δ est préalablement fixé, et dans lequel ladite valeur d'évaluation de la stationnarité est égale à ladite valeur de comparaison des projection $C[n]$ telle que la valeur de comparaison satisfait la condition de non-stationnarité des données : $C[n] > \delta$.

[Revendication 7] Dispositif (100), selon l'une des revendications 4 ou 5, dans lequel ledit seuil de non-stationnarité des données δ est ajusté en fonction d'une probabilité de fausse alarme fixée ζ, et dans lequel la détermination de ladite valeur d'évaluation de la stationnarité comprend la mise en œuvre d'un test statistique basé sur une détection a contrario de non-stationnarités des données.

[Revendication 8] Dispositif (100), selon la revendication 7, dans lequel ladite valeur d'évaluation de la stationnarité de ladite détection a contrario est dé-

terminée à partir d'une fonction de répartition F_E définie à partir de la valeur de comparaison des projections $C[n]$, la condition de non-stationnarité des données étant telle que :

$F_E(C[n]) = P(E > \delta | H0)$.

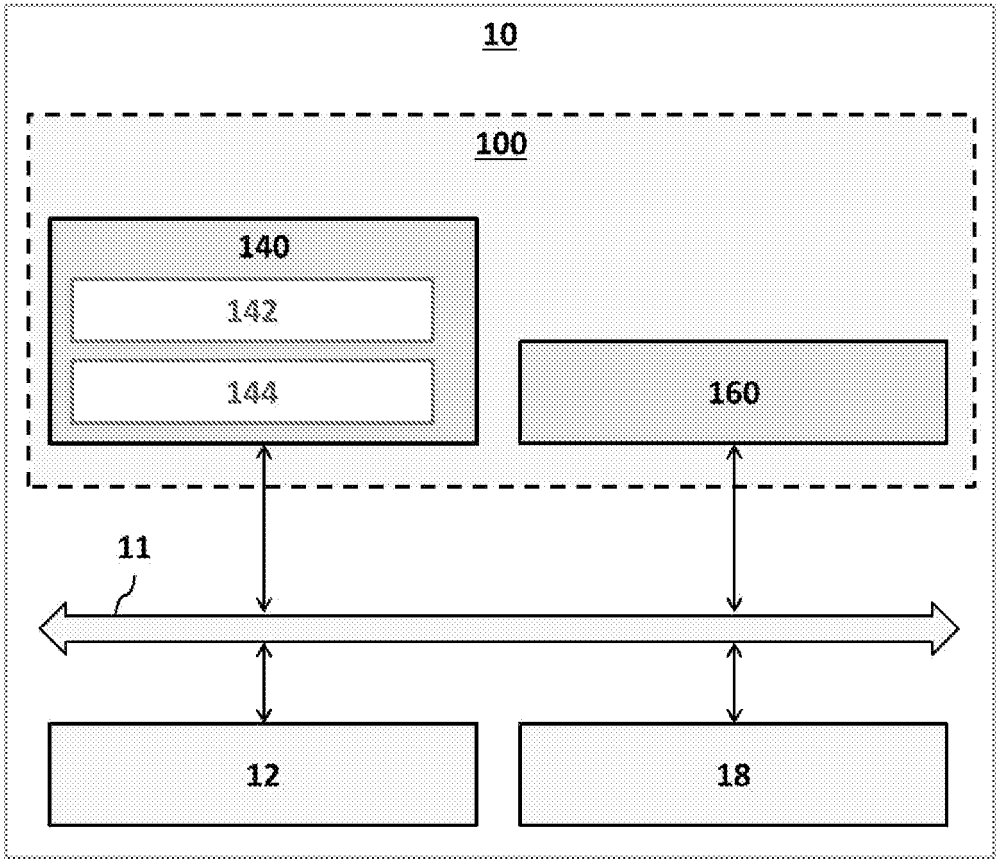
[Revendication 9] Système de construction de structures de partitionnement de données (10) comprenant au moins une source de données (12), le dispositif de partitionnement de données (100) selon l'une des revendications 1 à 8, et un module d'exploitation (18), dans lequel le module d'exploitation (18) est configuré pour déterminer au moins un score de pureté moyenne associé à une distribution de données d'une sous-structure $P[n]$ et pour évaluer la qualité de l'arbre de partitionnement de données à partir dudit au moins un score de pureté moyenne déterminé.

[Revendication 10] Procédé mis en œuvre par un dispositif de partitionnement de données (100), caractérisé en ce que le procédé comprend les étapes consistant à :

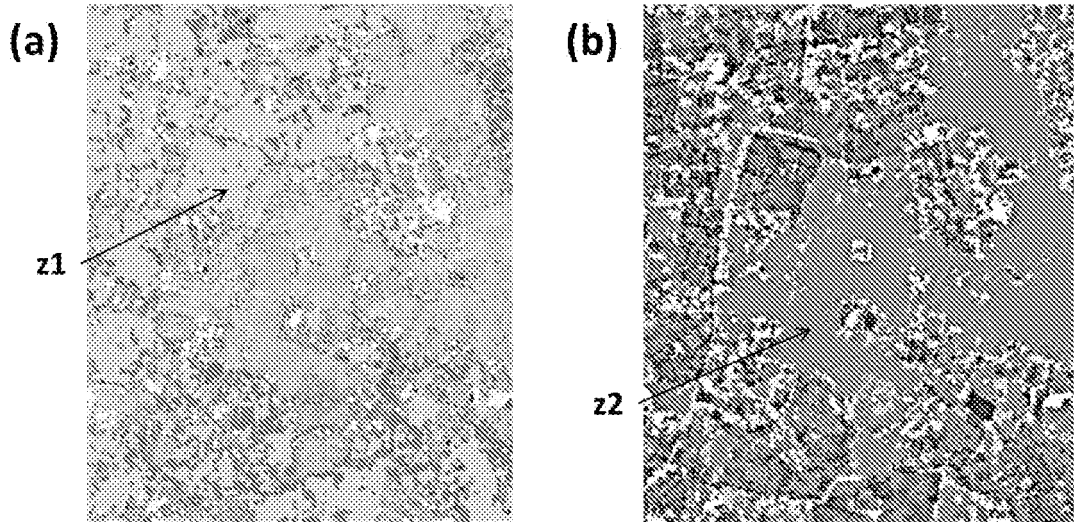
- Recevoir (600) un ensemble de données initial B_0 depuis au moins une source de données (12) ;
- Effectuer (606 à 620) au moins une opération de partitionnement initiale pour générer un arbre de partitionnement de données initial A_0 à partir dudit ensemble des données initial B_0, ledit arbre étant associé à N sous-structures $P[n]$ comprenant chacune une distribution de données de construction $B_0[n]$ issue de l'ensemble de données initial B_0 ;
- Recevoir (700) un ou plusieurs ensembles de données additionnels B_k depuis au moins une source de données (12) ;
- Effectuer (704 à 716) au moins une opération d'insertion pour insérer ledit ensemble de données additionnels B_k dans l'arbre de partitionnement de données initial A_0, lesdites sous-structures $P[n]$ comprenant en outre au moins une distribution de données insérée $B_k[n]$ issue de l'ensemble de données additionnels B_k ;
- Effectuer (802 à 806) au niveau de chaque sous-structure $P[n]$ une comparaison $C[n]$ de ladite distribution de données de construction $B_0[n]$ avec ladite distribution de données insérée $B_k[n]$ et déterminer une valeur d'évaluation de la stationnarité des données insérées par rapport aux données de construction de l'arbre de partitionnement de données initial A_0 à partir de ladite comparaison ; et
- Mettre à jour (808 à 818) ledit arbre de partitionnement de données à

partir de ladite valeur d'évaluation.

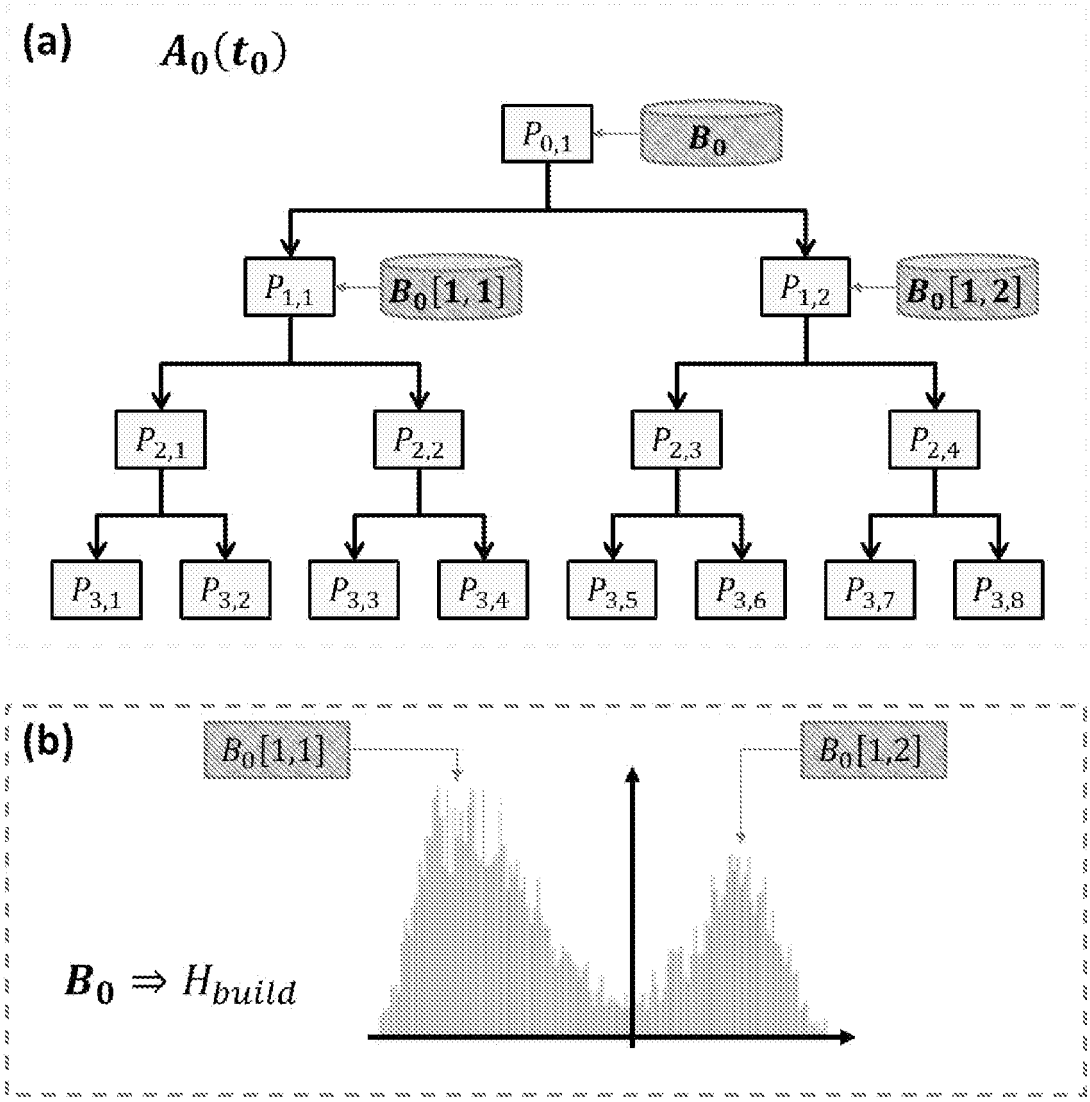
[Fig. 1]



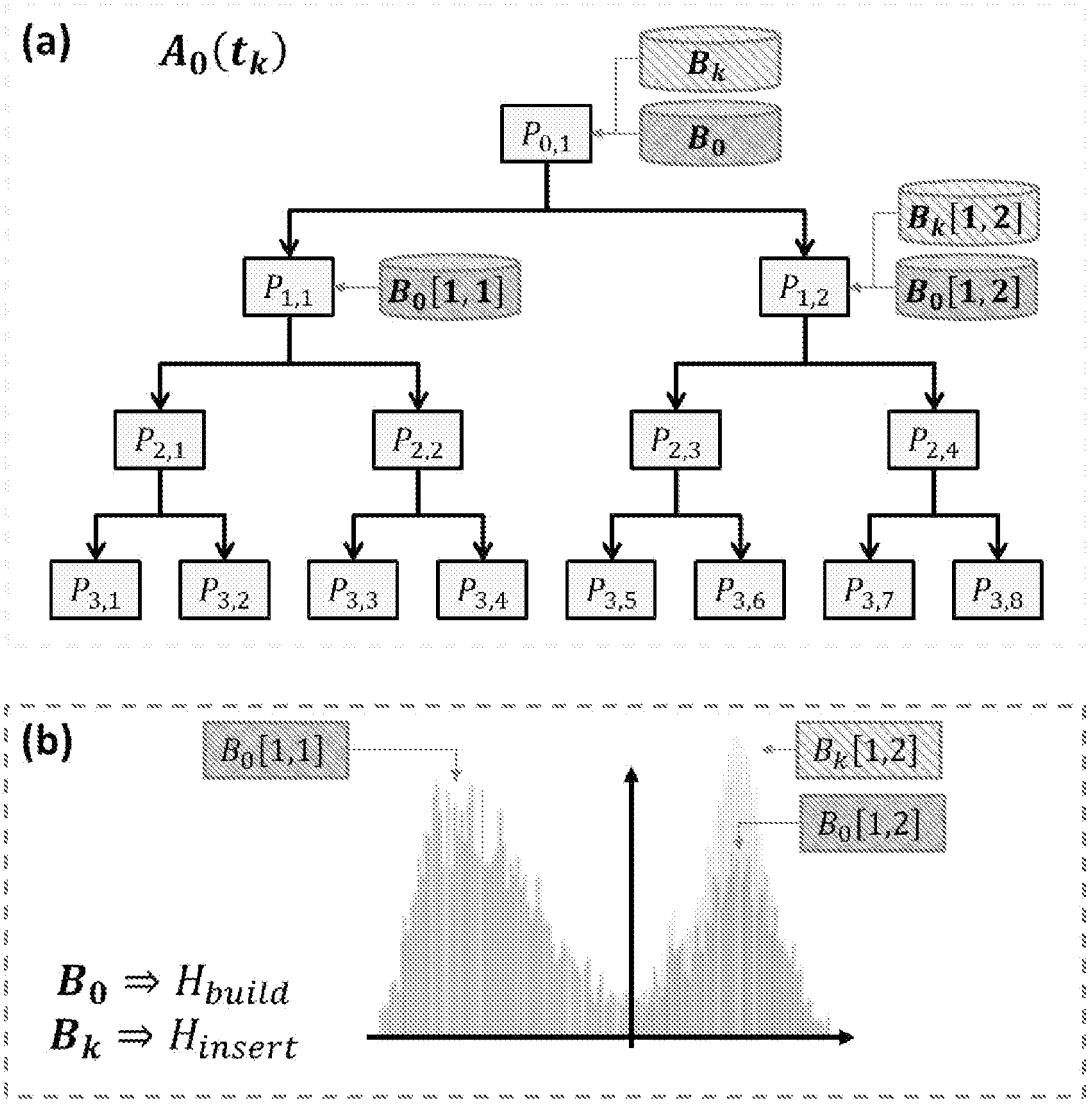
[Fig. 2]



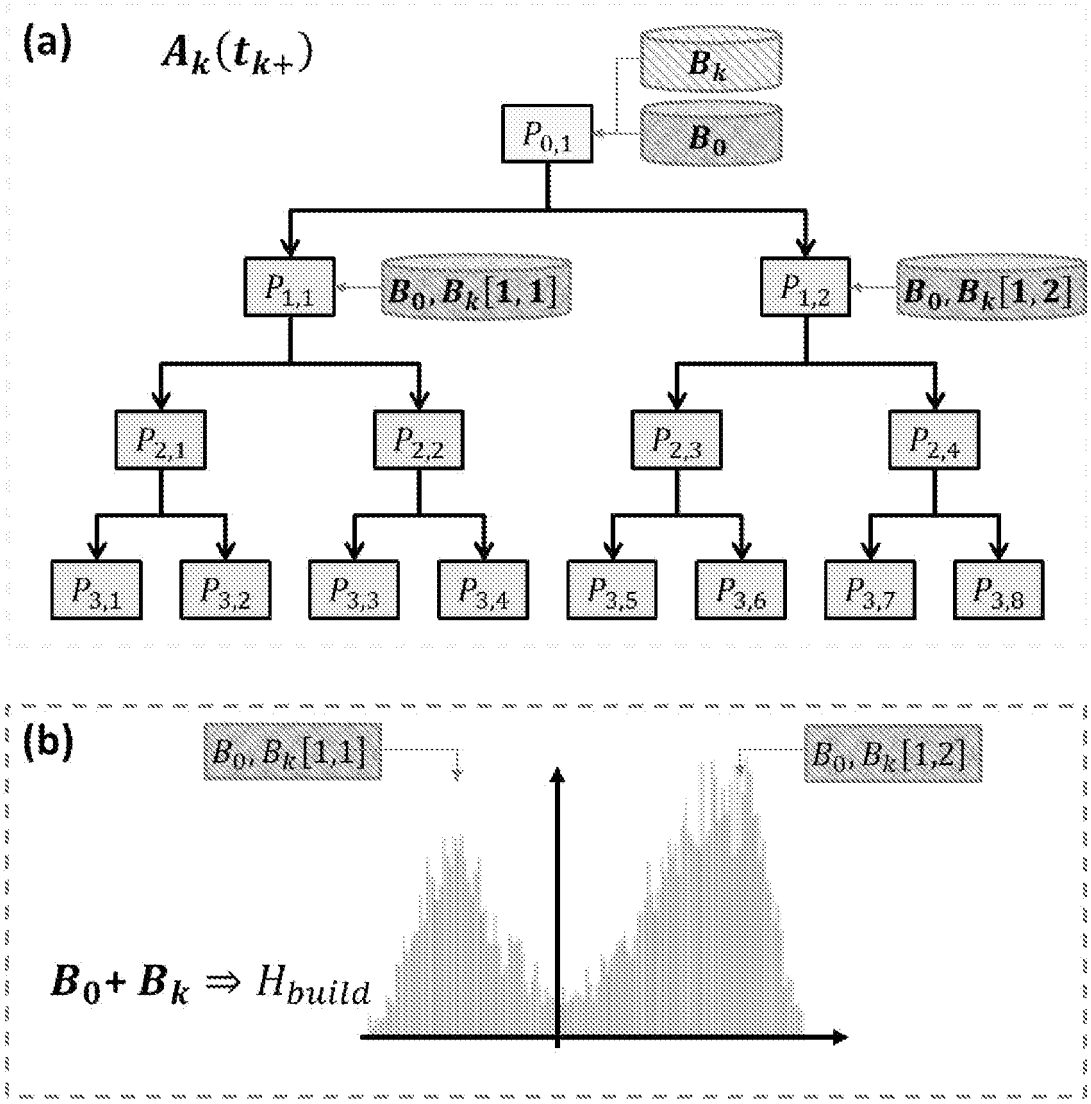
[Fig. 3]


(a)
$A_0(t_0)$
$P_{0,1}$
B_0
$P_{1,1}$
$B_0[1,1]$
$P_{1,2}$
$B_0[1,2]$
$P_{2,1}$
$P_{2,2}$
$P_{2,3}$
$P_{2,4}$
$P_{3,1}$
$P_{3,2}$
$P_{3,3}$
$P_{3,4}$
$P_{3,5}$
$P_{3,6}$
$P_{3,7}$
$P_{3,8}$
(b)
$B_0[1,1]$
$B_0[1,2]$
$B_0 \Rightarrow H_{build}$

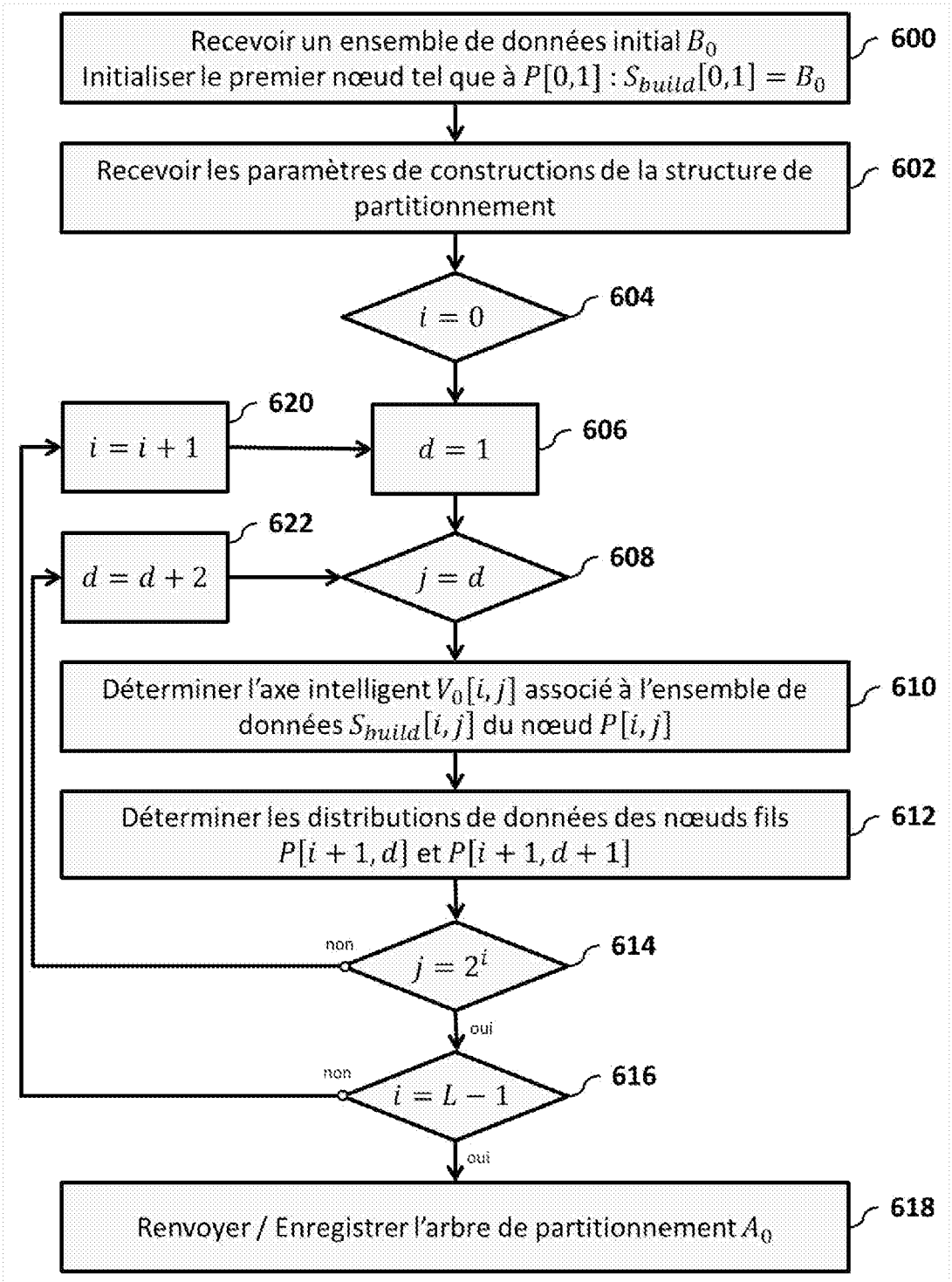
[Fig. 4]


(a)
$A_0(t_k)$
B_k
B_0
$P_{0,1}$
$B_k[1,2]$
$P_{1,1}$
$B_0[1,1]$
$P_{1,2}$
$B_0[1,2]$
$P_{2,1}$
$P_{2,2}$
$P_{2,3}$
$P_{2,4}$
$P_{3,1}$
$P_{3,2}$
$P_{3,3}$
$P_{3,4}$
$P_{3,5}$
$P_{3,6}$
$P_{3,7}$
$P_{3,8}$
(b)
$B_0[1,1]$
$B_k[1,2]$
$B_0[1,2]$
$B_0 \Rightarrow H_{build}$
$B_k \Rightarrow H_{insert}$

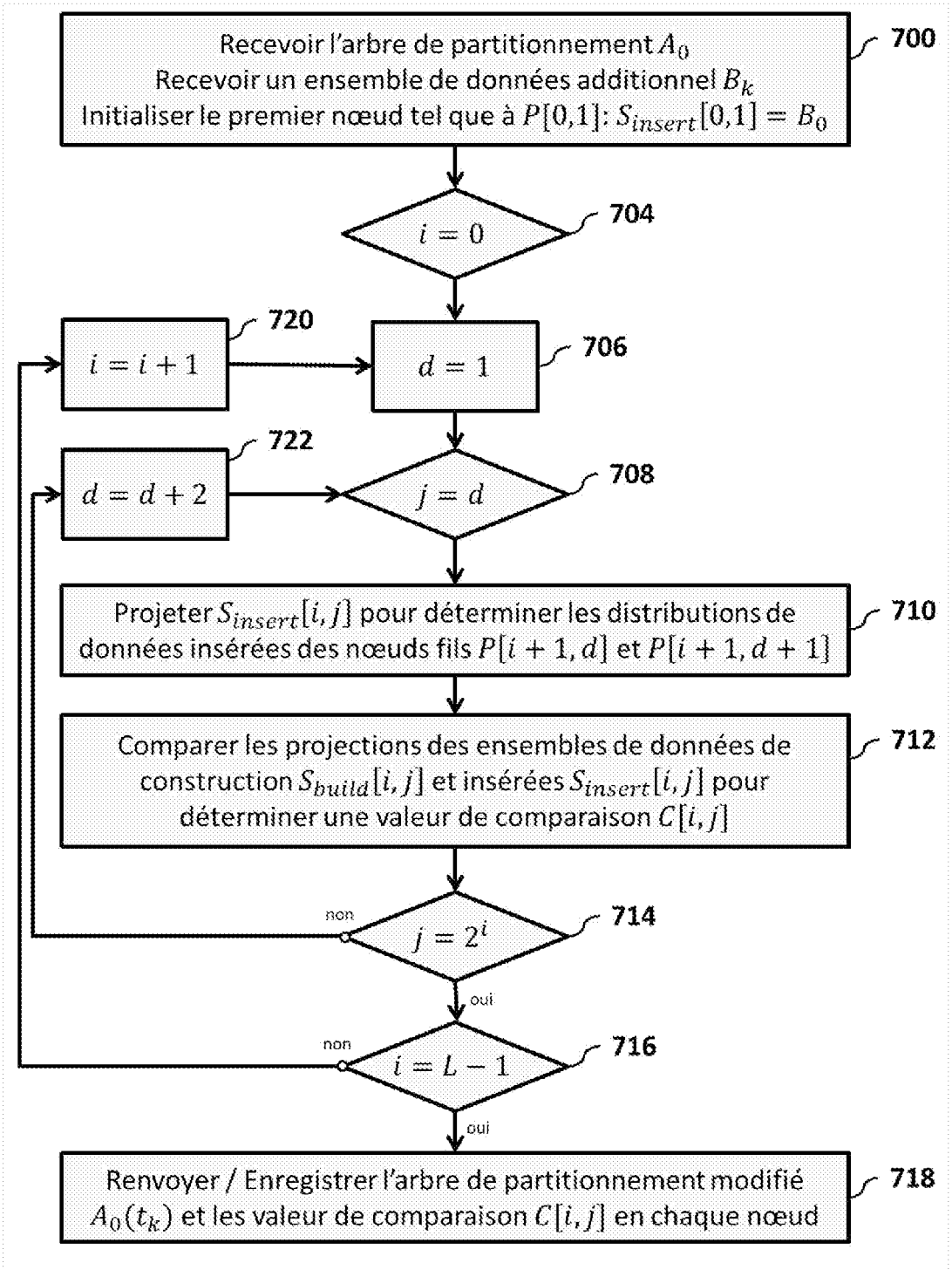
[Fig. 5]


(a)
$A_k(t_{k+})$
B_k
B_0
$P_{0,1}$
$P_{1,1}$
$B_0, B_k[1,1]$
$P_{1,2}$
$B_0, B_k[1,2]$
$P_{2,1}$
$P_{2,2}$
$P_{2,3}$
$P_{2,4}$
$P_{3,1}$
$P_{3,2}$
$P_{3,3}$
$P_{3,4}$
$P_{3,5}$
$P_{3,6}$
$P_{3,7}$
$P_{3,8}$
(b)
$B_0, B_k[1,1]$
$B_0, B_k[1,2]$
$B_0 + B_k \Rightarrow H_{build}$

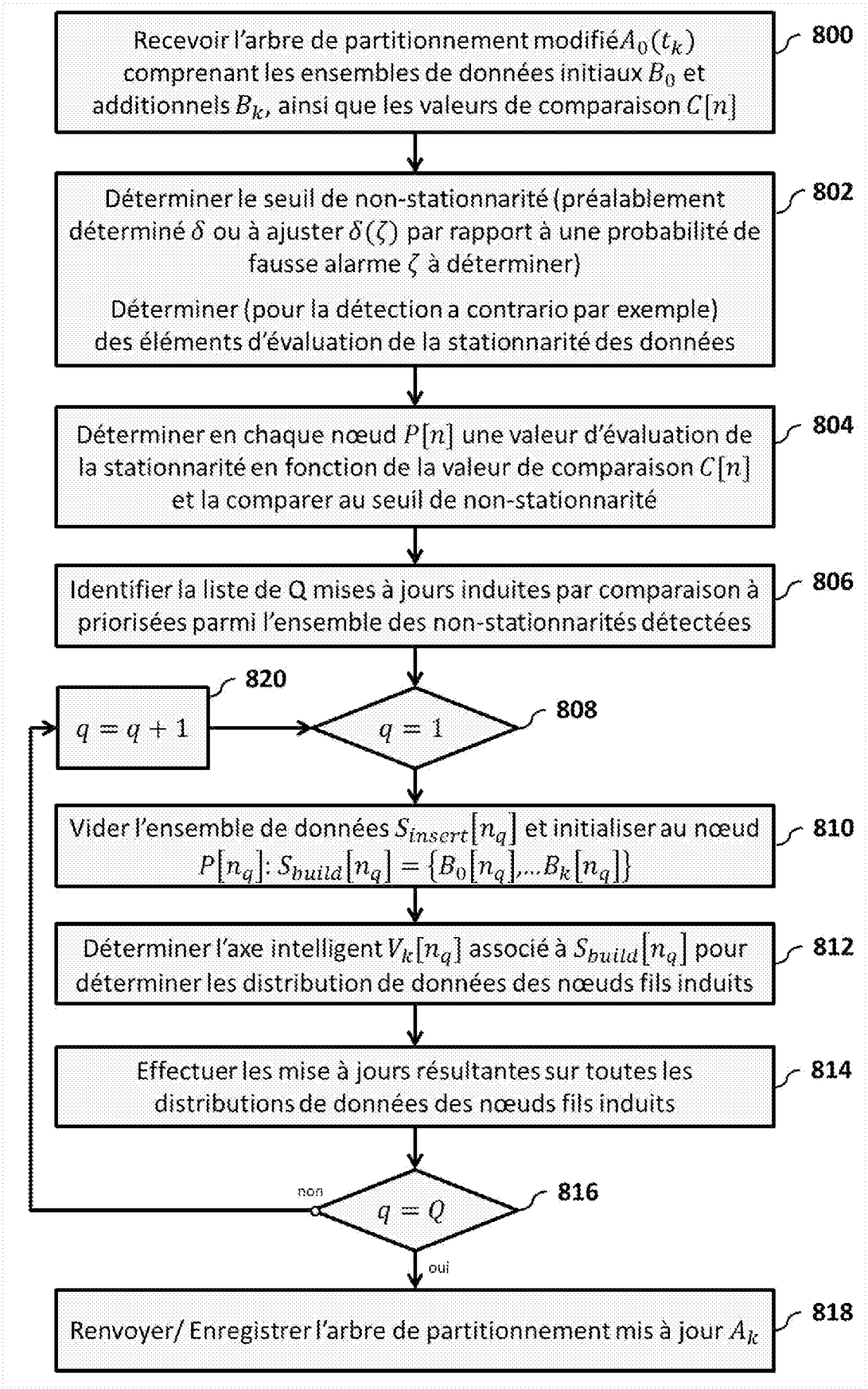
[Fig. 6]


Recevoir un ensemble de données initial B_0
Initialiser le premier nœud tel que à $P[0,1] : S_{build}[0,1] = B_0$
600
Recevoir les paramètres de constructions de la structure de partitionnement
602
$i = 0$
604
$i = i + 1$
620
$d = 1$
606
$d = d + 2$
622
$j = d$
608
Déterminer l'axe intelligent $V_0[i,j]$ associé à l'ensemble de données $S_{build}[i,j]$ du nœud $P[i,j]$
610
Déterminer les distributions de données des nœuds fils $P[i+1,d]$ et $P[i+1,d+1]$
612
non
$j = 2^i$
614
oui
non
$i = L - 1$
616
oui
Renvoyer / Enregistrer l'arbre de partitionnement A_0
618

[Fig. 7]


Recevoir l'arbre de partitionnement A_0
Recevoir un ensemble de données additionnel B_k
Initialiser le premier nœud tel que à $P[0,1]$: $S_{insert}[0,1] = B_0$
700
$i = 0$
704
$i = i + 1$
720
$d = 1$
706
$d = d + 2$
722
$j = d$
708
Projeter $S_{insert}[i,j]$ pour déterminer les distributions de données insérées des nœuds fils $P[i+1,d]$ et $P[i+1,d+1]$
710
Comparer les projections des ensembles de données de construction $S_{build}[i,j]$ et insérées $S_{insert}[i,j]$ pour déterminer une valeur de comparaison $C[i,j]$
712
non
$j = 2^i$
714
oui
non
$i = L - 1$
716
oui
Renvoyer / Enregistrer l'arbre de partitionnement modifié $A_0(t_k)$ et les valeur de comparaison $C[i,j]$ en chaque nœud
718

[Fig. 8]


Recevoir l'arbre de partitionnement modifié $A_0(t_k)$ comprenant les ensembles de données initiaux B_0 et additionnels B_k, ainsi que les valeurs de comparaison $C[n]$
800
Déterminer le seuil de non-stationnarité (préalablement déterminé δ ou à ajuster $\delta(\zeta)$ par rapport à une probabilité de fausse alarme ζ à déterminer)
Déterminer (pour la détection a contrario par exemple) des éléments d'évaluation de la stationnarité des données
802
Déterminer en chaque nœud $P[n]$ une valeur d'évaluation de la stationnarité en fonction de la valeur de comparaison $C[n]$ et la comparer au seuil de non-stationnarité
804
Identifier la liste de Q mises à jours induites par comparaison à priorisées parmi l'ensemble des non-stationnarités détectées
806
820
$q = q + 1$
$q = 1$
808
Vider l'ensemble de données $S_{insert}[n_q]$ et initialiser au nœud $P[n_q]$: $S_{build}[n_q] = \{B_0[n_q], \ldots B_k[n_q]\}$
810
Déterminer l'axe intelligent $V_k[n_q]$ associé à $S_{build}[n_q]$ pour déterminer les distribution de données des nœuds fils induits
812
Effectuer les mise à jours résultantes sur toutes les distributions de données des nœuds fils induits
814
non
$q = Q$
816
oui
Renvoyer/ Enregistrer l'arbre de partitionnement mis à jour A_k
818

RAPPORT DE RECHERCHE

articles L.612-14, L.612-53 à 69 du code de la propriété intellectuelle

OBJET DU RAPPORT DE RECHERCHE

L'I.N.P.I. annexe à chaque brevet un "RAPPORT DE RECHERCHE" citant les éléments de l'état de la technique qui peuvent être pris en considération pour apprécier la brevetabilité de l'invention, au sens des articles L. 611-11 (nouveauté) et L. 611-14 (activité inventive) du code de la propriété intellectuelle. Ce rapport porte sur les revendications du brevet qui définissent l'objet de l'invention et délimitent l'étendue de la protection.
Après délivrance, l'I.N.P.I. peut, à la requête de toute personne intéressée, formuler un "AVIS DOCUMENTAIRE" sur la base des documents cités dans ce rapport de recherche et de tout autre document que le requérant souhaite voir prendre en considération.

CONDITIONS D'ETABLISSEMENT DU PRESENT RAPPORT DE RECHERCHE

☐ Le demandeur a présenté des observations en réponse au rapport de recherche préliminaire.

[x] Le demandeur a maintenu les revendications.

☐ Le demandeur a modifié les revendications.

☐ Le demandeur a modifié la description pour en éliminer les éléments qui n'étaient plus en concordance avec les nouvelles revendications.

☐ Les tiers ont présenté des observations après publication du rapport de recherche préliminaire.

☐ Un rapport de recherche préliminaire complémentaire a été établi.

DOCUMENTS CITES DANS LE PRESENT RAPPORT DE RECHERCHE

La répartition des documents entre les rubriques 1, 2 et 3 tient compte, le cas échéant, des revendications déposées en dernier lieu et/ou des observations présentées.

☐ Les documents énumérés à la rubrique 1 ci-après sont susceptibles d'être pris en considération pour apprécier la brevetabilité de l'invention.

[x] Les documents énumérés à la rubrique 2 ci-après illustrent l'arrière-plan technologique général.

☐ Les documents énumérés à la rubrique 3 ci-après ont été cités en cours de procédure, mais leur pertinence dépend de la validité des priorités revendiquées.

☐ Aucun document n'a été cité en cours de procédure.

1. ELEMENTS DE L'ETAT DE LA TECHNIQUE SUSCEPTIBLES D'ETRE PRIS EN CONSIDERATION POUR APPRECIER LA BREVETABILITE DE L'INVENTION

NEANT

2. ELEMENTS DE L'ETAT DE LA TECHNIQUE ILLUSTRANT L'ARRIERE-PLAN TECHNOLOGIQUE GENERAL

FR 2 792 153 A1 (CANON KK [JP])
13 octobre 2000 (2000-10-13)

FR 2 842 672 A1 (INST NAT RECH INF AUTOMAT [FR]) 23 janvier 2004 (2004-01-23)

US 2009/164182 A1 (PEDERSEN STEIN INGE [NO] ET AL) 25 juin 2009 (2009-06-25)

MCNAMES J: "A FAST NEAREST-NEIGHBOR ALGORITHM BASED ON A PRINCIPAL AXIS SEARCH TREE",
IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE COMPUTER SOCIETY, USA,
vol. 23, no. 9,
2 septembre 2001 (2001-09-02), pages 964-976, XP001133374,
ISSN: 0162-8828, DOI: 10.1109/34.955110

EP 4 002 852 A1 (LG ELECTRONICS INC [KR])
25 mai 2022 (2022-05-25)

3. ELEMENTS DE L'ETAT DE LA TECHNIQUE DONT LA PERTINENCE DEPEND DE LA VALIDITE DES PRIORITES

NEANT