(19) RÉPUBLIQUE FRANÇAISE

**INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE**

COURBEVOIE

(11) **N° de publication :** ***3 143 155***

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

(21) **N° d'enregistrement national :** **22 13083**

(51) Int Cl[8] : ***G 06 F 8/61*** *(2023.01), G 06 F 15/16, 8/54*

# (12) DEMANDE DE BREVET D'INVENTION A1

(22) **Date de dépôt :** 09.12.22.

(30) **Priorité :**

(43) **Date de mise à la disposition du public de la demande :** 14.06.24 Bulletin 24/24.

(56) **Liste des documents cités dans le rapport de recherche préliminaire :** *Se reporter à la fin du présent fascicule*

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

( ) **Demande(s) d'extension :**

(71) **Demandeur(s) :** *ORANGE Société anonyme* — FR.

(72) **Inventeur(s) :** HOTEL Olivier, ROUDET Franck et EXCOFFIER David.

(73) **Titulaire(s) :** ORANGE Société anonyme.

(74) **Mandataire(s) :** VIDON BREVETS ET STRATEGIE.

## (54) Procédé de génération d'une application de traitement d'au moins un flux multimédia, dispositif et programme d'ordinateur associés.

(57) Procédé de génération d'une application de traitement d'au moins un flux multimédia, dispositif et programme d'ordinateur associés

L'invention concerne une solution de génération d'une application de traitement d'un flux multimédia comprenant un assemblage de fonctions de traitement élémentaire, ces fonctions de traitement élémentaire étant instanciées sur des nœuds de calcul. Un fichier de configuration est fourni à chaque fonction de traitement élémentaire afin de réaliser cet assemblage. Ainsi, un tel procédé permet de bénéficier des avantages des techniques dites de « low code » ou de « no code » tout en offrant la possibilité de sélectionner des fonctions élémentaires instanciées sur des nœuds de calcul adaptés.

FIGURE 4

FR 3 143 155 - A1

# Description

## Titre de l'invention : Procédé de génération d'une application de traitement d'au moins un flux multimédia, dispositif et programme d'ordinateur associés

### Domaine de l'invention

[0001] Le domaine de l'invention est celui des applications dites de vision par ordinateur dont le but est de traiter des flux multimédias. Plus précisément, l'invention concerne des applications de vision par ordinateur assemblées de manière graphique ou « *low code* ».

### Art antérieur et ses inconvénients

[0002] Classiquement, les applications de vision par ordinateur reposent sur un code informatique écrit par des spécialistes de langages informatiques. Ces codes permettent d'appeler des fonctions, elles aussi codées, appliquant un ou plusieurs traitements à des images constitutives du flux multimédia à traiter.

[0003] Cependant, de telles applications manquent de modularité car elles sont codées de manière monolithique, les fonctions appelées étant intégrées au sein même du code de l'application. Ainsi, toute modification apportée à l'une de ces fonctions peut nécessiter une réécriture partielle ou totale du code de l'application.

[0004] Avec le développement des technologies dites du numérique, les besoins en développement d'applications ont augmenté. Pour répondre à ce besoin grandissant, des techniques de génération d'applications nécessitant peu ou pas de compétences en codage informatique ont été développées. De telles techniques sont dite « *low code* » pour « peu de code » ou « *no code* » pour « pas de code ».

[0005] De telles techniques consistent en l'écriture de codes informatiques correspondant à des fonctions de traitement élémentaire. Ainsi, chaque fonction de traitement élémentaire correspond à un bloc fonctionnel, par exemple un bloc « reconnaissance faciale », un bloc « redimensionnement de l'image », etc.

[0006] Une personne souhaitant générer une application spécifique, telle que par exemple une application de vision par ordinateur, sélectionne plusieurs blocs fonctionnels appartenant à une bibliothèque de fonctions de traitement élémentaire mise à sa disposition, et les assemble de manière ordonnée afin de générer l'application permettant d'obtenir le traitement escompté.

[0007] Les différentes fonctions de traitement élémentaire étant développées indépendamment les unes des autres, une modification impactant l'une d'elles n'a pas de répercussion sur les autres fonctions de traitement élémentaire constitutives de l'application ce qui réduit les besoins en réécriture du code de l'application.

[0008] Un inconvénient de ces solutions est que les différents blocs fonctionnels proposés par un fournisseur de service de génération d'applications « *low code* » ou « *no code* » sont instanciés sur un même nœud de calcul. Or, selon les traitements effectués par les différents blocs fonctionnels sur les images constitutives du flux multimédia, le nœud de calcul sur lequel ces blocs fonctionnels sont instanciés peut ne pas disposer des ressources nécessaires à la bonne exécution de certaines fonctions de traitement élémentaire. Dans un tel cas de figure, l'application de vision par ordinateur n'offre pas une qualité de service satisfaisante.

[0009] Il existe donc un besoin d'une technique de génération d'application de traitement d'un flux multimédia ne présentant pas tout ou partie des inconvénients précités.

## Exposé de l'invention

[0010] L'invention répond à ce besoin en proposant un procédé de génération d'une application de traitement d'au moins un flux multimédia, ladite application comprenant un assemblage de fonctions de traitement élémentaire.

[0011] Un tel procédé est particulier en ce qu'il est mis en œuvre par un module de génération d'applications comprenant les étapes suivantes de :

- sélection de fonctions de traitement élémentaire destinées à être assemblées pour générer ladite application de traitement, une fonction de traitement élémentaire étant instanciée sur un nœud de calcul exécutant ladite fonction élémentaire,
- pour une fonction de traitement élémentaire sélectionnée donnée, association d'un fichier de configuration comprenant au moins un paramètre relatif à au moins une première fonction de traitement élémentaire générant des données destinées à être traitées par ladite fonction de traitement élémentaire donnée.

[0012] Une telle solution repose astucieusement sur la fourniture, dans un fichier de configuration, de paramètres permettant à une fonction de traitement élémentaire donnée d'identifier une ou plusieurs autres fonctions de traitement élémentaire auprès desquelles obtenir des données à traiter.

[0013] La fourniture de ces paramètres rend possible l'instanciation de ces fonctions de traitement élémentaire sur des nœuds de calculs distincts. Chaque fonction de traitement élémentaire peut être instanciée sur un nœud de calcul disposant des ressources qui répondent à ses besoins. Dans la suite du document, on entend par nœud de calcul tout équipement ou dispositif disposant des ressources nécessaires à l'exécution d'une fonction de traitement élémentaire. Ainsi, un nœud de calcul peut être une caméra, un dispositif d'affichage tel qu'un écran ou un smartphone, une machine virtuelle, un nœud de calcul d'une grappe de nœuds de type Kubernetes, un serveur, un ordinateur personnel, un processeur ou CPU (pour « Central Processing Unit »), une carte graphique ou GPU (pour « Graphical Processing Unit ») etc.

[0014] Un tel fichier de configuration peut être généré automatiquement par le module de génération, par exemple lorsque les paramètres relatifs à une fonction de traitement élémentaire sont associés à cette dernière dans la bibliothèque de blocs fonctionnels. Ainsi, la simple sélection des fonctions de traitement élémentaire à assembler déclenche la génération des fichiers de configuration correspondant.

[0015] Le fichier de configuration peut également être généré par un utilisateur du module de génération pour chaque fonction de traitement élémentaire. Une telle implémentation permet une plus grande flexibilité quant à l'ordonnancement des fonctions de traitement élémentaire entre elles puisque l'utilisateur a la possibilité de choisir quelles sont les entrées pour une fonction de traitement élémentaire donnée au lieu d'être contraint par les informations fournies par la bibliothèque de blocs fonctionnels.

[0016] Ainsi, un tel procédé permet de bénéficier des avantages des techniques dites de « low code » ou de « no code » tout en offrant la possibilité de sélectionner des fonctions élémentaires instanciées sur des nœuds de calcul adaptés. Un tel procédé permet également de mutualiser les fonctions de traitement élémentaire entre plusieurs applications de traitement de flux multimédia distinctes.

[0017] Le paramètre relatif à ladite première fonction de traitement élémentaire fourni par le fichier de configuration comprend entre autres :

- une adresse réseau dudit nœud de calcul sur lequel ladite première fonction de traitement élémentaire est instanciée,
- un port dudit nœud de calcul sur lequel ladite première fonction de traitement élémentaire est instanciée au travers duquel les données générées par ladite première fonction de traitement élémentaire sont destinées à être fournies,
- un identifiant de ladite première fonction de traitement élémentaire.

[0018] De telles informations permettent à la fonction de traitement élémentaire donnée d'identifier le nœud de calcul concerné et de pouvoir collecter les données appropriées auprès de ce dernier.

[0019] Ces informations sont également facilement identifiables et compréhensibles pour un utilisateur qui peut alors aisément les introduire lui-même dans un fichier de configuration.

[0020] Plus particulièrement, le paramètre relatif à ladite première fonction de traitement élémentaire constitue un argument de ladite fonction de traitement élémentaire sélectionnée donnée.

[0021] Le fichier de configuration comprend en outre, pour ladite fonction de traitement élémentaire sélectionnée donnée :

- un identifiant d'au moins un module de fourniture desdites données générées par ladite fonction de traitement élémentaire sélectionnée donnée.

[0022] Un tel module de fourniture appartient au groupe comprenant en outre :

- un port dudit nœud de calcul sur lequel ladite fonction de traitement élémentaire sélectionnée donnée est instanciée,
- un module d'affichage dudit flux multimédia,
- une mémoire.

[0023] Selon la position de la fonction de traitement élémentaire donnée dans l'assemblage de fonctions de traitement élémentaire constituant l'application, les données générées par son exécution sont destinées à être soit fournies en entrée d'une autre fonction de traitement élémentaire qui va leur appliquer un traitement supplémentaire, soit fournies à des modules d'affichage et/ou de stockage. De tels modules d'affichage et/ou de stockage peuvent être embarqués dans le nœud de calcul.

[0024] L'invention a également pour objet un procédé de traitement d'au moins un flux multimédia par une application de traitement comprenant un assemblage de fonctions de traitement élémentaire, une fonction de traitement élémentaire étant instanciée sur un nœud de calcul exécutant ladite fonction élémentaire.

[0025] Un tel procédé de traitement est particulier en ce qu'il est mis en œuvre par un nœud de calcul et qu'il comprenant une étape de :

- obtention de données contenues dans ledit flux multimédia à traiter,
- traitement, par ladite fonction de traitement élémentaire instanciée sur ledit nœud, desdites données obtenues,
- génération de métadonnées associées auxdites données traitées,
- fourniture desdites données traitées et desdites métadonnées générées.

Plus particulièrement, les données obtenues sont fournies par au moins une fonction de traitement élémentaire identifiée dans un fichier de configuration de ladite fonction de traitement élémentaire instanciée sur ledit nœud.

[0026] En effet, comme expliqué plus haut, les différentes fonctions de traitement élémentaire constituant l'application se voient fournir un fichier de configuration comprenant les informations permettant à une fonction de traitement élémentaire donnée d'identifier le nœud de calcul auprès duquel collecter les données qu'elle doit traiter. Ces informations étant des arguments de la fonction de traitement élémentaire donnée, lui sont fournies lors de la génération de l'application. Ainsi, dès la première exécution de l'application, chaque fonction de traitement élémentaire est en capacité de collecter les données qu'elle doit traiter.

[0027] Dans certains cas, les données obtenues sont acquises par le nœud de calcul lui-même.

[0028] Cela est par exemple le cas lorsque le nœud de calcul est embarqué dans une caméra qui capte une scène.

[0029] Les données obtenues peuvent être accompagnées par des métadonnées qui leur sont associées, lesdites métadonnées ayant également été générées par la fonction de

traitement élémentaire identifiée dans le fichier de configuration de ladite fonction de traitement élémentaire instanciée sur ledit nœud.

[0030] Ces métadonnées peuvent contenir des informations relatives au traitement précédemment effectué sur les images du flux multimédia.

[0031] De la même manière les métadonnées générées par ladite fonction de traitement élémentaire instanciée sur ledit nœud appartiennent au groupe comprenant en outre :

- des coordonnées d'une caméra ayant acquis le contenu du flux multimédia,
- un identifiant de ladite caméra,
- des coordonnées d'au moins un vecteur de mouvement calculé pour au moins une image constitutive du flux multimédia,
- au moins un identifiant d'au moins un objet détecté dans au moins une image constitutive du flux multimédia.

[0032] Ces métadonnées sont comprises dans un champ de données dudit au moins un flux, relatif à au moins une image fixe à traiter.

[0033] Les métadonnées sont par exemple comprises dans un champ AAPn ou COM du format JPEG ou encore dans I'un des champs iTXt, tEXt et zTXt du format PNG.

[0034] Ainsi, il n'est pas nécessaire de générer de nouveaux messages pour transmettre ces données supplémentaires. Cela facilite la mise en œuvre de la présente invention dans des systèmes déjà existants.

[0035] L'invention concerne également un module de génération d'une application de traitement d'au moins un flux multimédia, ladite application comprenant un assemblage de fonctions de traitement élémentaire.

[0036] Un tel module de génération d'applications comprend des moyens pour :

- sélectionner des fonctions de traitement élémentaire destinées à être assemblées pour générer ladite application de traitement, une fonction de traitement élémentaire étant instanciée sur un nœud de calcul exécutant ladite fonction élémentaire,
- pour une fonction de traitement élémentaire sélectionnée donnée, associer un fichier de configuration comprenant au moins un paramètre relatif à au moins une première fonction de traitement élémentaire générant des données destinées à être traitées par ladite fonction de traitement élémentaire donnée.

[0037] L'invention a encore pour objet un nœud de calcul sur lequel est instanciée au moins une fonction de traitement élémentaire comprise dans une application de traitement d'un flux multimédia.

[0038] Un tel nœud de calcul exécute ladite fonction élémentaire et comprend au moins un processeur configuré pour :

- obtenir des données contenues dans ledit flux multimédia à traiter,
- traiter, au moyen de ladite fonction de traitement élémentaire instanciée sur ledit nœud, lesdites données obtenues,

- générer des métadonnées associées auxdites données traitées,
- fournir lesdites données traitées et lesdites métadonnées générées.

[0039] L'invention concerne enfin des produits programme d'ordinateur comprenant des instructions de code de programme pour la mise en œuvre des procédés tels que décrits précédemment, lorsqu'ils sont exécutés par un processeur.

[0040] L'invention vise également un support d'enregistrement lisible par un ordinateur sur lequel sont enregistrés des programmes d'ordinateur comprenant des instructions de code de programme pour l'exécution des étapes des procédés selon l'invention tels que décrits ci-dessus.

[0041] Un tel support d'enregistrement peut être n'importe quelle entité ou dispositif capable de stocker les programmes. Par exemple, le support peut comporter un moyen de stockage, tel qu'une ROM, par exemple un CD ROM ou une ROM de circuit microélectronique, ou encore un moyen d'enregistrement magnétique, par exemple une clé USB ou un disque dur.

[0042] D'autre part, un tel support d'enregistrement peut être un support transmissible tel qu'un signal électrique ou optique, qui peut être acheminé via un câble électrique ou optique, par radio ou par d'autres moyens, de sorte que les programmes d'ordinateur qu'il contient sont exécutables à distance. Les programmes selon l'invention peuvent être en particulier téléchargés sur un réseau, par exemple le réseau Internet.

[0043] Alternativement, le support d'enregistrement peut être un circuit intégré dans lequel les programmes sont incorporés, le circuit étant adapté pour exécuter ou pour être utilisé dans l'exécution des procédés objets de l'invention précités.

## Liste des figures

[0044] D'autres buts, caractéristiques et avantages de l'invention apparaîtront plus clairement à la lecture de la description suivante, donnée à titre de simple exemple illustratif, et non limitatif, en relation avec les figures, parmi lesquelles :

[0045] [Fig.1] : cette figure représente un système dans lequel la présente invention est mise en œuvre,

[0046] [Fig.2] : cette figure représente les différentes étapes mises en œuvre pour générer une application de traitement d'un flux multimédia,

[0047] [Fig.3] : cette figure représente un premier exemple d'assemblage de fonctions de traitement élémentaire,

[0048] [Fig.4] : cette figure représente un deuxième exemple d'assemblage de fonctions de traitement élémentaire,

[0049] [Fig.5] : cette figure représente les différentes étapes mises en œuvre pour traiter un flux multimédia au moyen d'une application de traitement représentée à la [Fig.3] et obtenue selon le procédé décrit en référence à la [Fig.2],

[0050] [Fig.6] : cette figure représente un module de génération selon un mode de réalisation

de l'invention. Un module de génération est apte à mettre en œuvre le procédé de génération d'une application de traitement d'un flux multimédia selon la [Fig.2].

[0051] **Description détaillée de modes de réalisation de l'invention**

[0052] Le principe général de l'invention repose sur la génération d'une application de traitement d'un flux multimédia comprenant un assemblage de fonctions de traitement élémentaire, ces fonctions de traitement élémentaire étant instanciées sur des nœuds de calcul. Un fichier de configuration est fourni à chaque fonction de traitement élémentaire afin de réaliser cet assemblage.

[0053] Ainsi, un tel procédé permet de bénéficier des avantages des techniques dites de « *low code* » ou de « *no code* » tout en offrant la possibilité de sélectionner des fonctions élémentaires instanciées sur des nœuds de calcul adaptés.

[0054] On présente désormais, en relation avec la [Fig.1] un système dans lequel la présente invention est mise en œuvre.

[0055] Un tel système est un système distribué dans lequel une pluralité de nœuds de calculs communiquent entre eux pour traiter un flux multimédia.

[0056] Le système ainsi représenté comprend, par exemple, un module de génération 10 d'application de traitement d'un flux multimédia. Un tel module de génération 10 est par exemple embarqué dans un serveur ou dans une machine virtuelle opéré(e) par un fournisseur de services de traitements de flux multimédia.

[0057] Un tel module de génération 10 comprend une bibliothèque 12 de fonctions de traitement élémentaire FTE, ou blocs fonctionnels. De telles fonctions de traitement élémentaire FTE sont par exemple une fonction d'acquisition d'images représentatives d'une scène, une fonction de redimensionnement d'une image, une fonction de colorimétrie, une fonction de détection d'objets, une fonction d'affichage d'un contenu multimédia ou de stockage d'un contenu multimédia, etc. Chacune de ces fonctions de traitement élémentaire FTE est instanciée sur un nœud de calcul 11-i disposant des ressources nécessaires à l'exécution de la fonction de traitement élémentaire FTE instanciée. Un même nœud de calcul peut instancier plusieurs fonctions de traitement élémentaire FTE. A titre d'exemple, un nœud de calcul peut être tout équipement ou dispositif disposant des ressources nécessaires à l'exécution d'une fonction de traitement élémentaire. Ainsi, un nœud de calcul peut être une caméra, un dispositif d'affichage tel qu'un écran ou un smartphone, une machine virtuelle, un nœud de calcul d'une grappe de nœuds de type Kubernetes, un serveur, un ordinateur personnel, un CPU, un GPU, etc.

[0058] Bien que quatre nœuds de calculs 11-1 à 11-4 soient représentés à la [Fig.1], le nombre de nœuds de calculs impliqués dans la mise en œuvre de la présente solution dépend du nombre de fonctions de traitement élémentaire FTE assemblées pour constituer l'application de traitement d'un flux multimédia.

[0059] Dans la présente solution, trois types de fonctions de traitement élémentaire FTE sont envisagés. Un premier type de fonctions de traitement élémentaire est qualifié de source FTES, un deuxième type est qualifié d'opération FTEO et un troisième type est qualifié de puit FTEP.

[0060] Une fonction source FTES est une fonction de traitement élémentaire n'ayant pas d'entrée mais au moins une sortie.

[0061] Lorsqu'une fonction de traitement élémentaire n'a pas d'entrée, cela signifie que les données qu'elle traite ne lui sont pas fournies par une autre fonction de traitement élémentaire mais qu'elle les obtient par un autre biais. Une fonction source FTES est par exemple une fonction de décodage d'images acquises par une caméra ou une fonction de dématriçace (ou « *demosaicing* » en langue anglaise. Une fonction source FTES peut également lire les données qu'elle doit traiter depuis une mémoire.

[0062] Une sortie d'une fonction de traitement élémentaire est par exemple un port du nœud de calcul 11-i sur lequel elle est instanciée et via lequel les données générées par la fonction de traitement élémentaire sont fournies (ou rendues accessibles).

[0063] Une fonction d'opération FTEO est une fonction de traitement intermédiaire ayant une ou plusieurs entrées et une ou plusieurs sorties. Une telle fonction d'opération FTEO obtient des données à traiter sur chacune de ses entrées à partir d'une sortie d'une autre fonction de traitement élémentaire FTE source ou d'opération et rend accessibles les données qu'elle a générées sur chacune de ses sorties.

[0064] Enfin, une fonction puits FTEP est une fonction de traitement élémentaire n'ayant pas de sortie mais au moins une entrée.

[0065] Lorsqu'une fonction de traitement élémentaire n'a pas de sortie, cela signifie que les données qu'elle a traitées ne sont pas fournies à une autre fonction de traitement élémentaire mais qu'elles sont présentées à un utilisateur final. Une fonction puits FTEP est par exemple un dispositif d'affichage ou un module de stockage des données.

[0066] Une application de traitement d'un flux multimédia est un assemblage d'au moins une fonction source FTES et d'au moins une fonction puits FTEP. Une telle application peut également comprendre au moins une fonction d'opération FTEO.

[0067] La [Fig.2] représente les différentes étapes mises en œuvre pour générer une application de traitement d'un flux multimédia.

[0068] Dans une étape E1, le module de génération 10 sélectionne, dans la bibliothèque 12, au moins une fonction source FTES, dans l'exemple décrit il s'agit de la fonction source « Camera », au moins une fonction d'opération FTEO, dans l'exemple décrit deux fonctions d'opérations sont sélectionnées : une première fonction d'opération « Resize » qui redimensionne la taille des images et une deuxième fonction d'opération « YOLO » (pour « *You Only Look Once* » en anglais) qui détecte des objets dans une image, et au moins une fonction puits FTEP « Viz » qui permet d'afficher le résultat

des différents traitements appliqués sur un écran. Bien entendu, d'autres fonctions de traitement élémentaire peuvent être sélectionnées et assemblées afin de générer une application de traitement d'un flux multimédia.

[0069] Un utilisateur final peut interagir avec le module de génération 10 au moyen d'une interface homme-machine (non représenté sur les figures) telle qu'un écran tactile ou un couple écran-clavier afin de transmettre au module de génération 10 des instructions permettant de sélectionner les différentes fonctions élémentaires FTE à assembler.

[0070] Une fois les différentes fonctions de traitement élémentaires FTE sélectionnées, ces dernières sont assemblées afin de générer l'application de traitement d'un flux multimédia.

[0071] Pour cela, un fichier de configuration FC est fourni à chacune des fonctions de traitement élémentaire FTE sélectionnées.

[0072] Un tel fichier de configuration FC est généré au cours d'une étape E2. Dans un premier mode de réalisation, un fichier de configuration FC peut être généré automatiquement par le module de génération 10, par exemple lorsque les paramètres relatifs à une fonction de traitement élémentaire sont associés à cette dernière dans la bibliothèque 12. Ainsi, la sélection et l'ordonnancement entre elles des fonctions de traitement élémentaire FTE à assembler déclenche la génération du ou des fichiers de configuration correspondants.

[0073] Dans un deuxième mode de réalisation, le fichier de configuration FC peut également être généré par l'utilisateur du module de génération 10. Une telle implémentation permet une plus grande flexibilité quant à l'ordonnancement des fonctions de traitement élémentaire entre elles puisque l'utilisateur a la possibilité de choisir quelles sont les entrées pour une fonction de traitement élémentaire donnée au lieu d'être contraint par les informations fournies par la bibliothèque 12.

[0074] Pour une fonction source FTES, un tel fichier de configuration FC comprend au moins un identifiant d'au moins une sortie ou module de fourniture sur laquelle les données traitées ou générées par la fonction source FTES sont destinées à être rendues accessibles.

[0075] Un tel identifiant est par exemple un couple port/path, ou « port » correspondant à un numéro de port du nœud de calcul sur lequel la fonction FTE est instanciée et « path » est un paramètre identifiant la fonction de traitement élémentaire FTE ayant rendu les données accessibles. Le paramètre « port » est particulièrement utile lorsqu'un même nœud de calcul instancie plusieurs fonctions de traitement élémentaire FTE.

[0076] Pour une fonction d'opération FTEO, un tel fichier de configuration FC comprend au moins un identifiant d'au moins une entrée et au moins un identifiant d'au moins une sortie sur laquelle les données traitées ou générées par la fonction d'opération FTEO sont destinées à être fournies.

[0077] Plus particulièrement, un tel fichier de configuration FC comprend pour chaque entrée de la fonction d'opération FTEO une adresse réseau d'un nœud de calcul sur lequel au moins une fonction source FTS et/ou au moins une autre fonction d'opération sont instanciées, ces fonctions de traitement élémentaire fournissant des données à traiter à la fonction d'opération FTEO, un port des nœuds de calcul concernés sur lequel ces fonctions source FTES et/ou ces fonctions d'opérations FTEO fournissent les données générées et un identifiant de ces fonctions source FTES et/ou de ces fonctions d'opérations FTEO.

[0078] Le fichier de configuration FC comprend également au moins un identifiant d'au moins une sortie sur laquelle les données traitées ou générées par la fonction d'opération FTEO sont destinées à être fournies.

[0079] Enfin, pour une fonction puits FTEP, un tel fichier de configuration FC comprend pour chaque entrée de la fonction puits FTEP une adresse réseau d'un nœud de calcul sur lequel au moins une fonction source FTS et/ou au moins une fonction d'opération sont instanciées, ces fonctions de traitement élémentaire fournissant des données à traiter à la fonction puits FTEP, un port des nœuds de calcul concernés sur lequel ces fonctions source FTES et/ou ces fonctions d'opérations FTEO fournissent les données générées et un identifiant de ces fonctions source FTES et/ou de ces fonctions d'opérations FTEO.

[0080] Une fois les fichiers de configuration générés, ces derniers sont transmis dans une étape E3 aux différentes fonctions de traitement élémentaire FTE sélectionnées.

[0081] La [Fig.3] représente un premier exemple d'assemblage de fonctions de traitement élémentaire FTE. Un tel assemblage comprend une fonction source FTES « Camera » configurée pour décoder les images acquises par une caméra. Cette fonction source présente une unique sortie. Une telle fonction source est par exemple instanciée sur un nœud de calcul 11-1 de type CPU.

[0082] L'assemblage comprend ensuite une première fonction d'opération « Resize » qui re-dimensionne les images acquises par la caméra. La fonction « Resize » a une entrée qui collecte les données fournies par la fonction « Camera » et une unique sortie. Une telle fonction d'opération est par exemple instanciée sur un nœud de calcul 11-2 de type CPU.

[0083] L'assemblage comprend une deuxième fonction d'opération « YOLO » qui détecte et identifie les objets présents dans les images acquises par la caméra. La fonction « YOLO » a une unique entrée qui collecte les données fournies par la fonction « Resize » et une unique sortie. Une telle fonction d'opération est par exemple instanciée sur un nœud de calcul 11-3 de type GPU.

[0084] Enfin, l'assemblage comprend une fonction puits FTEP « Viz » qui affiche les images acquises par la caméra sur un écran. La fonction « Viz » a une unique entrée

qui collecte les données fournies par la fonction « YOLO » et ne comprend aucune sortie. Une telle fonction puits est par exemple instanciée sur un nœud de calcul 11-4 de type CPU.

[0085] En référence à l'assemblage décrit à la [Fig.3], le fichier de configuration FC1 de la fonction « Camera » comprend les données suivantes :

[0086] /Camera -port=8080,

[0087] où /Camera est l'identifiant de la fonction source, et -port=8080 l'identifiant du module de fourniture sur lequel les données traitées par la fonction puit sont rendues accessibles.

[0088] Le fichier de configuration FC2 de la fonction « Resize » comprend les données suivantes :

[0089] /Resize 192.168.0.1 : 8080/cam -port=8080,

[0090] où /Resize est l'identifiant de la première fonction d'opération, 192.168.0.1 est l'adresse réseau du nœud de calcul 11-1 sur lequel la fonction « Camera » est instanciée et 8080/cam le couple port/path du nœud de calcul sur lequel la fonction « Camera » est instanciée et sur lequel les données traitées par la fonction « Camera » sont fournies, et -port=8080 l'identifiant du module de fourniture sur lequel les données traitées par la fonction « Resize » sont rendues accessibles.

[0091] Le fichier de configuration FC3 de la fonction « YOLO » comprend les données suivantes :

[0092] /YOLO 192.168.5.1 : 8080/rsz -port=8080,

[0093] où /YOLO est l'identifiant de la deuxième fonction d'opération, 192.168.5.1 est l'adresse réseau du nœud de calcul 11-2 sur lequel la fonction « Resize » est instanciée et 8080/rsz le couple port/path du nœud de calcul sur lequel la fonction « Resize » est instanciée et sur lequel les données traitées par la fonction « Resize » sont fournies, et -port=8080 l'identifiant du module de fourniture sur lequel les données traitées par la fonction « YOLO » sont rendues accessibles.

[0094] Enfin, le fichier de configuration FC4 de la fonction « Viz » comprend les données suivantes :

[0095] /Viz 192.168.10.1 : 8080/objs,

[0096] où /Viz est l'identifiant de la deuxième fonction puits, 192.168.10.1 est l'adresse réseau du nœud de calcul 11-3 sur lequel la fonction « YOLO » est instanciée et 8080/objs le couple port/path du nœud de calcul sur lequel la fonction « YOLO » est instanciée et sur lequel les données traitées par la fonction « YOLO » sont fournies.

[0097] Une fois en possession de son fichier de configuration FC, chaque fonction de traitement élémentaire FTE sait auprès de quel nœud de calcul obtenir les données qu'elle doit traiter et sur quel module d'accès elle doit rendre accessible ses propres données.

[0098] La [Fig.4] représente un deuxième exemple d'assemblage de fonctions de traitement élémentaire FTE. Un tel assemblage comprend une fonction source FTES « Camera » configurée pour décoder les images acquises par une caméra. Cette fonction source présente une unique sortie. Une telle fonction source est par exemple instanciée sur un nœud de calcul 11-1 de type CPU.

[0099] L'assemblage comprend ensuite une première fonction d'opération « Resize » qui re-dimensionne les images acquises par la caméra. La fonction « Resize » a une entrée qui collecte les données fournies par la fonction « Camera » et deux sorties. Une telle fonction d'opération est par exemple instanciée sur un nœud de calcul 11-2 de type CPU.

[0100] L'assemblage comprend une deuxième fonction d'opération « YOLO » qui détecte et identifie les objets présents dans les images acquises par la caméra. La fonction « YOLO » a une unique entrée qui collecte les données fournies par la fonction « Resize » et une unique sortie. Une telle fonction d'opération est par exemple instanciée sur un nœud de calcul 11-3 de type GPU.

[0101] L'assemblage comprend une troisième fonction d'opération « Face » qui détecte par exemple les visages présents dans les images acquises par la caméra. La fonction « Face » a une unique entrée qui collecte les données fournies par la fonction « Resize » et une unique sortie. Une telle fonction d'opération est par exemple instanciée sur un nœud de calcul 11-5 de type GPU.

[0102] Enfin, l'assemblage comprend une fonction puits FTEP « Viz » qui affiche les images acquises par la caméra sur un écran. La fonction « Viz » a deux entrées, une première entrée qui collecte les données fournies par la fonction « YOLO » et une deuxième entrée qui collecte les données fournies par la fonction « Face » et ne contient aucune sortie. Une telle fonction puits est par exemple instanciée sur un nœud de calcul 11-4 de type CPU.

[0103] En référence à l'assemblage décrit à la [Fig.4], le fichier de configuration FC1 de la fonction « Camera » comprend les données suivantes :

[0104] /Camera -port=8080,

[0105] où /Camera est l'identifiant de la fonction source, et -port=8080 l'identifiant du module de fourniture sur lequel les données traitées par la fonction puit sont rendues accessibles.

[0106] Le fichier de configuration FC2 de la fonction « Resize » comprend les données suivantes :

[0107] /Resize 192.168.0.1 : 8080/cam -port=8080,

[0108] où /Resize est l'identifiant de la première fonction d'opération, 192.168.0.1 est l'adresse réseau du nœud de calcul 11-1 sur lequel la fonction « Camera » est instanciée et 8080/cam le couple port/path du nœud de calcul sur lequel la fonction

« Camera » est instanciée et sur lequel les données traitées par la fonction « Camera » sont fournies, et -port=8080 l'identifiant du module de fourniture sur lequel les données traitées par la fonction « Resize » sont rendues accessibles.

[0109] Le fichier de configuration FC3 de la fonction « YOLO » comprend les données suivantes :

[0110] /YOLO 192.168.5.1 : 8080/rsz -port=8080,

[0111] où /YOLO est l'identifiant de la deuxième fonction d'opération, 192.168.5.1 est l'adresse réseau du nœud de calcul 11-2 sur lequel la fonction « Resize » est instanciée et 8080/rsz le couple port/path du nœud de calcul sur lequel la fonction « Resize » est instanciée et sur lequel les données traitées par la fonction « Resize » sont fournies, et -port=8080 l'identifiant du module de fourniture sur lequel les données traitées par la fonction « YOLO » sont rendues accessibles.

[0112] Le fichier de configuration FC5 de la fonction « Face » comprend les données suivantes :

[0113] /Face 192.168.5.1 : 8080/rsz -port=8080,

[0114] où /Face est l'identifiant de la deuxième fonction d'opération, 192.168.5.1 est l'adresse réseau du nœud de calcul 11-2 sur lequel la fonction « Resize » est instanciée et 8080/rsz le couple port/path du nœud de calcul sur lequel la fonction « Resize » est instanciée et sur lequel les données traitées par la fonction « Resize » sont fournies, et -port=8080 l'identifiant du module de fourniture sur lequel les données traitées par la fonction « Face » sont rendues accessibles.

[0115] Enfin, le fichier de configuration FC4 de la fonction « Viz » comprend les données suivantes :

[0116] /Viz 192.168.10.1 : 8080/objs,

[0117] /Viz 192.168.15.1 :8080/fcs

[0118] où /Viz est l'identifiant de la deuxième fonction puits, 192.168.10.1 est l'adresse réseau du nœud de calcul 11-3 sur lequel la fonction « YOLO » est instanciée et 8080/objs le couple port/path du nœud de calcul sur lequel la fonction « YOLO » est instanciée et sur lequel les données traitées par la fonction « YOLO » sont fournies, et 192.168.15.1 est l'adresse réseau du nœud de calcul 11-5 sur lequel la fonction « Face » est instanciée et 8080/fcs le couple port/path du nœud de calcul sur lequel la fonction « Face » est instanciée et sur lequel les données traitées par la fonction « Face » sont fournies.

[0119] Une fois en possession de son fichier de configuration FC, chaque fonction de traitement élémentaire FTE sait auprès de quel nœud de calcul obtenir les données qu'elle doit traiter et sur quel module de fourniture elle doit rendre accessible ses propres données.

[0120] Les données comprises dans ces différents fichiers de configuration FC sont

considérées comme des arguments des fonctions de traitement élémentaire FTE qui leur sont fournis en préambule de leur exécution.

[0121] La [Fig.5] représente les différentes étapes mises en œuvre pour traiter un flux multimédia au moyen de l'application de traitement représentée à la [Fig.3] et obtenue selon le procédé décrit en référence à la [Fig.2].

[0122] Dans une étape G1, une fonction source, ici la fonction « Camera », acquiert des images d'une scène qu'elle décode au cours d'une étape G2.

[0123] La fonction « Camera » génère également des métadonnées relatives aux images décodées. De telles métadonnées sont par exemple, un identifiant de la caméra, une résolution de cette dernière, un horodatage des images capturées, une géolocalisation des images capturées, etc. De telles métadonnées sont comprises par exemple dans le champ AAPn (pour « *Application Specific* ») ou le champ COM (pour « *Comment* ») du format JPEG (pour « Joint Photographic Experts Group ») dans lequel les images sont codées. D'autres formats de traitement d'images fixes et leurs champs de données correspondants peuvent bien entendu être utilisés, tels que par exemple le format PNG avec entre autres les champs iTXt, tEXt et zTXt, etc.

[0124] Conformément à ce qui est indiqué dans le fichier de configuration FC, la fonction « Camera » fournit les images décodées et leurs métadonnées sur le port 8080 du nœud de calcul 11-1 sur lequel la fonction « Camera » est instanciée dans une étape G3.

[0125] Au cours d'une étape G4, la fonction « Resize » obtient, conformément aux informations comprises dans son fichier de configuration FC1, les données fournies par la fonction « Camera ».

[0126] A l'aide des informations comprises dans les métadonnées accompagnant les données obtenues, la fonction « Resize » traite ces dernières dans une étape G5. La fonction « Resize » génère également des métadonnées relatives aux images redimensionnées, ces métadonnées pouvant toujours comprendre les informations générées par la fonction « Camera », telles que les informations d'horodatage par exemple.

[0127] Conformément à ce qui est indiqué dans le fichier de configuration FC1, la fonction « Resize » fournit les images redimensionnées et leurs métadonnées sur le port 8080 du nœud de calcul 11-2 sur lequel la fonction « Resize » est instanciée dans une étape G6.

[0128] Au cours d'une étape G7, la fonction « YOLO » obtient, conformément aux informations comprises dans son fichier de configuration FC2, les données fournies par la fonction « Resize ».

[0129] A l'aide des informations comprises dans les métadonnées accompagnant les données obtenues, la fonction « YOLO » traite ces dernières au cours d'une étape G8. La fonction « YOLO » génère également des métadonnées relatives aux objets identifiés dans les images redimensionnées, ces métadonnées pouvant toujours comprendre les informations générées par la fonction « Camera » et la fonction

« Resize », telle que les informations d'horodatage par exemple, ainsi que des informations relatives aux objets identifiés telles que leur nom, leur position dans l'image, etc.

[0130] Conformément à ce qui est indiqué dans le fichier de configuration FC2, la fonction « YOLO » fournit les images redimensionnées et leurs métadonnées sur le port 8080 du nœud de calcul 11-3 sur lequel la fonction « YOLO » est instanciée dans une étape G9.

[0131] Au cours d'une étape G10, la fonction « Viz » obtient, conformément aux informations comprises dans son fichier de configuration FC, les données fournies par la fonction « YOLO ».

[0132] A l'aide des informations comprises dans les métadonnées accompagnant les données obtenues, la fonction « Viz » traite ces dernières dans une étape G11 et les images ainsi traitées sont ainsi affichées sur un écran.

[0133] La [Fig.6] représente un module de génération 10 selon un mode de réalisation de l'invention. Un module de génération 10 est apte à mettre en œuvre le procédé de génération d'une application de traitement d'un flux multimédia selon la [Fig.2].

[0134] Un module de génération 10 peut comprendre au moins un processeur matériel 61, une unité de stockage 62, au moins une interface 63, et au moins une interface homme-machine 64 qui sont connectés entre eux au travers d'un bus 65. Bien entendu, les éléments constitutifs du module de génération 10 peuvent être connectés au moyen d'une connexion autre qu'un bus.

[0135] Le processeur 61 commande les opérations du module de génération 10. L'unité de stockage 62 stocke au moins un programme pour la mise en œuvre du procédé selon un mode de réalisation de l'invention à exécuter par le processeur 61, et diverses données, telles que des paramètres utilisés pour des calculs effectués par le processeur 61, des données intermédiaires de calculs effectués par le processeur 61, etc. Le processeur 61 peut être formé par tout matériel ou logiciel connu et approprié, ou par une combinaison de matériel et de logiciel. Par exemple, le processeur 61 peut être formé par un matériel dédié tel qu'un circuit de traitement, ou par une unité de traitement programmable telle qu'une unité centrale de traitement (*Central Processing Unit*) qui exécute un programme stocké dans une mémoire de celui-ci.

[0136] L'unité de stockage 62 peut être formée par n'importe quel moyen approprié capable de stocker le programme ou les programmes et des données d'une manière lisible par un ordinateur. Des exemples d'unité de stockage 62 comprennent des supports de stockage non transitoires lisibles par ordinateur tels que des dispositifs de mémoire à semi-conducteurs, et des supports d'enregistrement magnétiques, optiques ou magnéto-optiques chargés dans une unité de lecture et d'écriture.

[0137] L'interface 63 fournit une connexion entre le module de génération 10 et l'ensemble

des nœuds de calcul 11-i du système.

[0138] L'interface homme-machine 64 peut être un écran tactile ou un ensemble comprenant un clavier, une souris et un écran.

# Revendications

[Revendication 1] Procédé de génération d'une application de traitement d'au moins un flux multimédia, ladite application comprenant un assemblage de fonctions de traitement élémentaire, ledit procédé étant mis en œuvre par un module de génération d'application et comprenant les étapes suivantes :
- sélection de fonctions de traitement élémentaire destinées à être assemblées pour générer ladite application de traitement, une fonction de traitement élémentaire étant instanciée sur un nœud de calcul exécutant ladite fonction élémentaire,
- pour une fonction de traitement élémentaire sélectionnée donnée, association d'un fichier de configuration comprenant au moins un paramètre relatif à au moins une première fonction de traitement élémentaire générant des données destinées à être traitées par ladite fonction de traitement élémentaire donnée.

[Revendication 2] Procédé de génération d'une application de traitement selon la revendication 1 dans lequel ledit paramètre relatif à ladite première fonction de traitement élémentaire comprend:
- une adresse réseau dudit nœud de calcul sur lequel ladite première fonction de traitement élémentaire est instanciée,
- un port dudit nœud de calcul sur lequel ladite première fonction de traitement élémentaire est instanciée, au travers duquel les données générées par ladite première fonction de traitement élémentaire sont destinées à être fournies,
- un identifiant de ladite première fonction de traitement élémentaire.

[Revendication 3] Procédé de génération d'une application de traitement selon la revendication 2 dans lequel ledit paramètre relatif à ladite première fonction de traitement élémentaire est un argument de ladite fonction de traitement élémentaire sélectionnée donnée.

[Revendication 4] Procédé de génération d'une application de traitement selon les revendications 1 à 3 dans lequel ledit fichier de configuration comprend en outre, pour ladite fonction de traitement élémentaire sélectionnée donnée :
- un identifiant d'au moins un module de fourniture desdites données générées par ladite fonction de traitement élémentaire sélectionnée donnée.

[Revendication 5] Procédé de génération d'une application de traitement selon la reven-

dication 4 dans lequel ledit identifiant est un argument de ladite fonction de traitement élémentaire sélectionnée donnée.

[Revendication 6] Procédé de génération d'une application de traitement selon la revendication 4 ou la revendication 5 dans lequel ledit module appartient au groupe comprenant:
- un port dudit nœud de calcul sur lequel ladite fonction de traitement élémentaire sélectionnée donnée est instanciée,
- un module d'affichage dudit flux multimédia,
- une mémoire.

[Revendication 7] Procédé de traitement d'au moins un flux multimédia par une application de traitement comprenant un assemblage de fonctions de traitement élémentaire, une fonction de traitement élémentaire étant instanciée sur un nœud de calcul exécutant ladite fonction élémentaire, ledit procédé étant mis en œuvre par un nœud de calcul et comprenant ce qui suit :
- obtention de données contenues dans ledit flux multimédia à traiter,
- traitement, par ladite fonction de traitement élémentaire instanciée sur ledit nœud, desdites données obtenues,
- génération de métadonnées associées auxdites données traitées,
- fourniture desdites données traitées et desdites métadonnées générées.

[Revendication 8] Procédé de traitement d'au moins un flux multimédia selon la revendication 7 dans lequel les données obtenues sont fournies par au moins une fonction de traitement élémentaire identifiée dans un fichier de configuration de ladite fonction de traitement élémentaire instanciée sur ledit nœud.

[Revendication 9] Procédé de traitement d'au moins un flux multimédia selon la revendication 8 dans lequel le nœud de calcul obtient des métadonnées associées auxdites données obtenues, lesdites métadonnées ayant également été générées par la fonction de traitement élémentaire identifiée dans le fichier de configuration de ladite fonction de traitement élémentaire instanciée sur ledit nœud.

[Revendication 10] Procédé de traitement d'au moins un flux multimédia selon la revendication 7 dans lequel les données obtenues sont acquises par ledit nœud de calcul.

[Revendication 11] Procédé de traitement d'au moins un flux multimédia selon l'une quelconque des revendications 7 à 10 dans lequel les métadonnées générées par ladite fonction de traitement élémentaire instanciée sur ledit nœud appartiennent au groupe comprenant :

- des coordonnées d'une caméra ayant acquis un contenu du flux multimédia,
- un identifiant de ladite caméra,
- des coordonnées d'au moins un vecteur de mouvement calculé pour au moins une image contenue dans le flux multimédia,
- au moins un identifiant d'au moins un objet détecté dans au moins une image contenue dans le flux multimédia.

[Revendication 12] Procédé de traitement d'au moins un flux multimédia selon l'une quelconque des revendications 7 à 11 dans lequel les métadonnées sont comprises dans un champ de données dudit au moins un flux, relatif à au moins une image fixe à traiter.

[Revendication 13] Module de génération d'une application de traitement d'au moins un flux multimédia, ladite application comprenant un assemblage de fonctions de traitement élémentaire, ledit module de génération d'applications comprenant des moyens pour :
- sélectionner des fonctions de traitement élémentaire destinées à être assemblées pour générer ladite application de traitement, une fonction de traitement élémentaire étant instanciée sur un nœud de calcul exécutant ladite fonction élémentaire,
- pour une fonction de traitement élémentaire sélectionnée donnée, associer un fichier de configuration comprenant au moins un paramètre relatif à au moins une première fonction de traitement élémentaire générant des données destinées à être traitées par ladite fonction de traitement élémentaire donnée.

[Revendication 14] Nœud de calcul sur lequel est instanciée au moins une fonction de traitement élémentaire comprise dans une application de traitement d'un flux multimédia, ledit nœud de calcul exécutant ladite fonction élémentaire et comprenant au moins un processeur configuré pour :
- obtenir des données contenues dans ledit flux multimédia à traiter,
- traiter, au moyen de ladite fonction de traitement élémentaire instanciée sur ledit nœud, lesdites données obtenues,
- générer des métadonnées associées auxdites données traitées,
- fournir lesdites données traitées et lesdites métadonnées générées.

[Fig. 1]

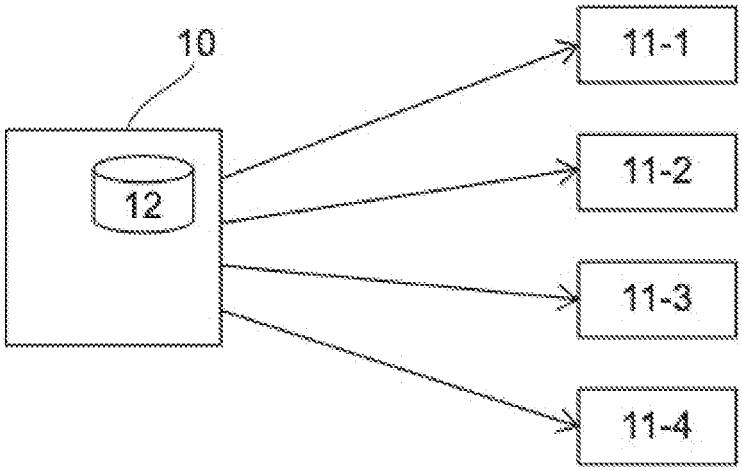



Fig. 1

[Fig. 2]

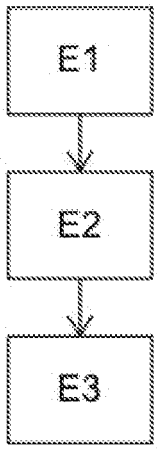



Fig. 2

[Fig. 3]

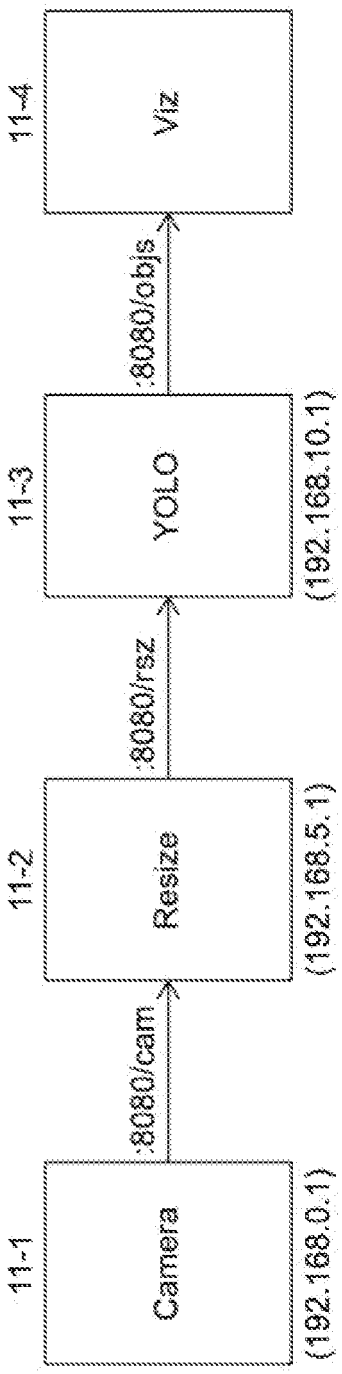

11-1
Camera
(192.168.0.1)
:8080/cam
11-2
Resize
(192.168.5.1)
:8080/rsz
11-3
YOLO
(192.168.10.1)
:8080/objs
11-4
Viz

Fig. 3

[Fig. 4]

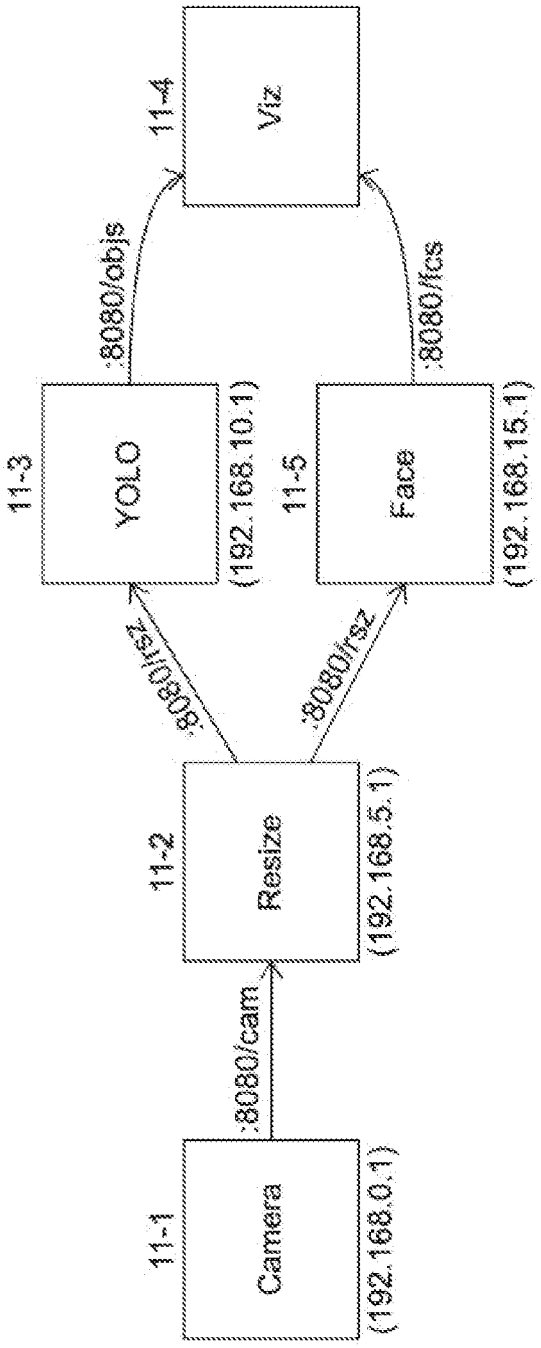

11-1
Camera
(192.168.0.1)
:8080/cam
11-2
Resize
(192.168.5.1)
:8080/rsz
:8080/rsz
11-3
YOLO
(192.168.10.1)
11-5
Face
(192.168.15.1)
:8080/objs
:8080/fcs
11-4
Viz

Fig. 4

[Fig. 5]

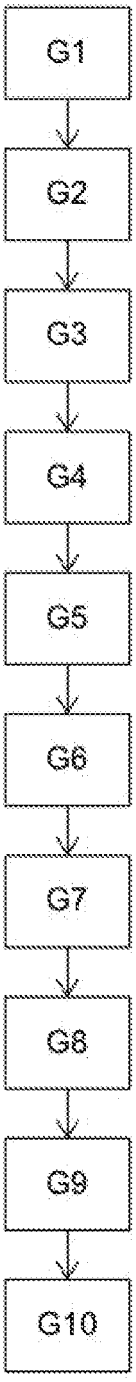



Fig. 5

[Fig. 6]

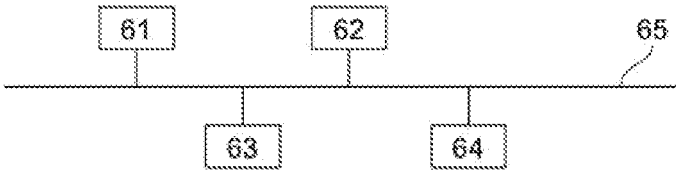



Fig. 6

RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE

# RAPPORT DE RECHERCHE PRÉLIMINAIRE PARTIEL

établi sur la base des dernières revendications déposées avant le commencement de la recherche

voir FEUILLE(S) SUPPLÉMENTAIRE(S)

N° d'enregistrement national

FA 913746
FR 2213083

| DOCUMENTS CONSIDÉRÉS COMME PERTINENTS | | Revendications concernées | Classement attribué à l'invention par l'INPI |
|---|---|---|---|
| Catégorie | Citation du document avec indication, en cas de besoin, des parties pertinentes | | |
| X | US 2019/114152 A1 (COLLINS KEITH [US]) 18 avril 2019 (2019-04-18) * alinéas [0002], [0008], [0065], [0051] * | 1-6,13 | G06F8/61 G06F15/16 G06F8/54 |
| A | US 2008/005752 A1 (MORRIS ROBERT PAUL [US]) 3 janvier 2008 (2008-01-03) * alinéas [0035], [0036], [0063], [0064] * * abrégé * | 1-6,13 | |
| A | US 2003/037119 A1 (AUSTIN PAUL F [US]) 20 février 2003 (2003-02-20) * abrégé * * alinéas [0012], [0013], [0032] * | 1-6,13 | |
| A | US 2020/153744 A1 (BRADDY RICK GENE [US] ET AL) 14 mai 2020 (2020-05-14) * abrégé; revendication 1 * * alinéa [0077] * | 1-6,13 | |
| A | US 2005/039162 A1 (CIFRA CHRISTOPHER G [US]) 17 février 2005 (2005-02-17) * abrégé; revendication 1 * * alinéa [0282] * | 1-6,13 | DOMAINES TECHNIQUES RECHERCHÉS (IPC) G06F |
| A | US 2002/059401 A1 (AUSTIN PAUL F [US]) 16 mai 2002 (2002-05-16) * abrégé; revendication 1 * * alinéa [0088] * | 1-6,13 | |
| A | US 2002/070965 A1 (AUSTIN PAUL F [US]) 13 juin 2002 (2002-06-13) * abrégé; revendication 1 * * alinéas [0049], [0197] * | 1-6,13 | |

-/--

| Date d'achèvement de la recherche | Examinateur |
|---|---|
| 26 mai 2023 | Hoareau, Samuel |

CATÉGORIE DES DOCUMENTS CITES

X : particulièrement pertinent à lui seul
Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie
A : arrière-plan technologique
O : divulgation non-écrite
P : document intercalaire

T : théorie ou principe à la base de l'invention
E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure.
D : cité dans la demande
L : cité pour d'autres raisons
.....
& : membre de la même famille, document correspondant

1

EPO FORM 1503 12.99 (P04C35)

RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE

# RAPPORT DE RECHERCHE PRÉLIMINAIRE PARTIEL

établi sur la base des dernières revendications déposées avant le commencement de la recherche

voir FEUILLE(S) SUPPLÉMENTAIRE(S)

N° d'enregistrement national

FA 913746
FR 2213083

| DOCUMENTS CONSIDÉRÉS COMME PERTINENTS | | Revendications concernées | Classement attribué à l'invention par l'INPI |
|---|---|---|---|
| Catégorie | Citation du document avec indication, en cas de besoin, des parties pertinentes | | |
| A | US 2009/037875 A1 (JONES ANDREW R [US] ET AL) 5 février 2009 (2009-02-05)<br>* abrégé; revendications 1,2 *<br>* alinéas [0009], [0025], [0026], [0035] *<br>----- | 1-6,13 | |

DOMAINES TECHNIQUES RECHERCHÉS (IPC)

| Date d'achèvement de la recherche | Examinateur |
|---|---|
| 26 mai 2023 | Hoareau, Samuel |

CATÉGORIE DES DOCUMENTS CITES

X : particulièrement pertinent à lui seul
Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie
A : arrière-plan technologique
O : divulgation non-écrite
P : document intercalaire

T : théorie ou principe à la base de l'invention
E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure.
D : cité dans la demande
L : cité pour d'autres raisons
.....
& : membre de la même famille, document correspondant

1

EPO FORM 1503 12.99 (P04C35)

## ABSENCE D'UNITÉ D'INVENTION
## FEUILLE SUPPLÉMENTAIRE B

Numéro de la demande

FA 913746
FR 2213083

La division de la recherche estime que la présente demande de brevet ne satisfait pas à l'exigence relative à l'unité d'invention et concerne plusieurs inventions ou pluralités d'inventions, à savoir :

1. **revendications: 1-6, 13**

   **Le premier groupe d'inventions, tel que défini par les revendications 1-6, 13, concerne l'association de fichiers configurations à des fonctions sélectionnées.**

   **---**

2. **revendications: 7-12, 14**

   **Le second groupe d'inventions, tel que défini par les revendications 7-12, 14, concerne le traitement et envoi de données par une fonction.**

   **---**

**La première invention a été recherchée.**

# ANNEXE AU RAPPORT DE RECHERCHE PRÉLIMINAIRE RELATIF A LA DEMANDE DE BREVET FRANÇAIS NO. FR 2213083 FA 913746

La présente annexe indique les membres de la famille de brevets relatifs aux documents brevets cités dans le rapport de recherche préliminaire visé ci-dessus.
Les dits membres sont contenus au fichier informatique de l'Office européen des brevets à la date du 26-05-2023
Les renseignements fournis sont donnés à titre indicatif et n'engagent pas la responsabilité de l'Office européen des brevets, ni de l'Administration française

| Document brevet cité au rapport de recherche | Date de publication | Membre(s) de la famille de brevet(s) | Date de publication |
|---|---|---|---|
| US 2019114152 A1 | 18-04-2019 | US 2018081639 A1 | 22-03-2018 |
| | | US 2018081640 A1 | 22-03-2018 |
| | | US 2019114152 A1 | 18-04-2019 |
| US 2008005752 A1 | 03-01-2008 | AUCUN | |
| US 2003037119 A1 | 20-02-2003 | AUCUN | |
| US 2020153744 A1 | 14-05-2020 | US 2018254989 A1 | 06-09-2018 |
| | | US 2020153744 A1 | 14-05-2020 |
| | | US 2021243134 A1 | 05-08-2021 |
| US 2005039162 A1 | 17-02-2005 | US 2005035966 A1 | 17-02-2005 |
| | | US 2005035967 A1 | 17-02-2005 |
| | | US 2005039160 A1 | 17-02-2005 |
| | | US 2005039162 A1 | 17-02-2005 |
| | | US 2005039170 A1 | 17-02-2005 |
| | | US 2012089935 A1 | 12-04-2012 |
| US 2002059401 A1 | 16-05-2002 | US 6370569 B1 | 09-04-2002 |
| | | US 2002059401 A1 | 16-05-2002 |
| | | US 2003200280 A1 | 23-10-2003 |
| US 2002070965 A1 | 13-06-2002 | AUCUN | |
| US 2009037875 A1 | 05-02-2009 | AUCUN | |

EPO FORM P0465