

Adaptation en temps réel pour une meilleure qualité d'expérience en réalité augmentée

THÈSE N° 4838 (2010)

PRÉSENTÉE LE 29 OCTOBRE 2010

À LA FACULTÉ SCIENCES ET TECHNIQUES DE L'INGÉNIEUR

LABORATOIRE D'AUTOMATIQUE

PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Damien PERRITAZ

acceptée sur proposition du jury:

Dr A. Karimi, président du jury
Dr D. Gillet, Dr C. Salzmann, directeurs de thèse
Prof. O. D. Crisalle, rapporteur
Dr F. Mondada, rapporteur
Dr K. Szafnicki, rapporteur



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2010

Thèse financée par la Fondation Hasler
Programme Man-Machine Interaction
Projet #1970 (6th Sense : Wearable Supervision System)

Remerciements

Ce document met un terme à quatre ans et demi de recherche académique au sein du Laboratoire d'Automatique. Des émotions nombreuses et diverses m'ont traversé durant cette période. Je voudrais vous faire part ici de l'une d'entre elles : la reconnaissance.

Comment remercier toutes ces personnes en évitant les fautes d'orthographe dans leur nom ? Comment faut-il faire pour n'oublier personne ? J'ai peut-être trouvé une astuce, en nommant les gens uniquement par la première lettre de leur nom...

Je tiens tout d'abord à remercier mes directeurs de thèse, D et C, pour leur soutien, particulièrement lors des dernières semaines de ma thèse. Merci d'avoir passé tout ce temps à lire mes documents, stylo rouge à la main, pour me permettre de m'améliorer. Merci de m'avoir guidé sur ce long chemin parfois brumeux.

D'autres personnes ont leur nom sur la couverture de ce document, il s'agit des membres du jury qui ont accepté de juger mon travail. Je remercie A d'avoir présidé le jury, et les rapporteurs Prof. C, Dr M et Dr S pour leurs critiques. Merci au Dr S pour sa lecture très approfondie, au Dr M pour ses remarques techniques judicieuses et au Prof. C pour la touchante discussion du lendemain de mon examen.

Je remercie mes collègues liés au *Projet 6^e Sens* qui a été le déclencheur de mon sujet de doctorat. Merci également à la *Fondation Hasler* d'avoir assuré le financement de ma recherche durant ces années.

VIII

Le Laboratoire d'Automatique est une grande famille, avec des individus qui arrivent et qui partent. Il y en a aussi certains qui restent, au fond du couloir. J'en profite pour remercier la tête du laboratoire qui gère efficacement la famille : Prof. L et Prof. B, ainsi que D et A.

Dans cette grande famille, il y a toujours quelqu'un vers qui s'orienter pour obtenir des compléments d'information théoriques. Merci donc au personnel scientifique actuel et passé. Je tiens à remercier spécialement P pour ses conseils et son intérêt.

Si la famille fonctionne, c'est aussi grâce au personnel administratif et technique, avec actuellement R, F, S et F. La famille a encore de nombreux autres membres, trop nombreux pour être cités. Mais je pense ne pas trop me tromper en remerciant mes futurs ex-collègues de A à Z (si si, il y en a même un dont le prénom commence par la dernière lettre de l'alphabet !) pour les bons moments passés à leurs côtés.

Au sein de la grande famille, il y a une petite fratrie composée des occupants de longue date du bureau *conciergerie*. J'ai eu le plaisir d'y connaître des personnages très attachants, bien au delà des limites de cette pièce tantôt studieuse, tantôt divertissante. Merci à B, D, K et Y pour leur support, leurs rires, leurs jeux et leur camaraderie.

Heureusement, il y a une vie à côté du travail. De nombreuses personnes m'ont aidé à garder l'équilibre nécessaire. Pour cela, je remercie sincèrement ma famille et mes amis. Je tiens à faire un petit clin d'oeil à ceux qui sont liés à mon travail. Je dois mon introduction sur la réalité augmentée à mes coéquipiers de hockey, auxquels j'ai expliqué mon travail au sommet d'une tour. L'orthographe et la grammaire de ma dissertation ont été améliorés grâce à mes relecteurs A, A et M. Je remercie également ma femme A, cobaye du prototype de réalité augmentée et soutien indispensable.

Finalement, merci à vous lecteurs qui donnez un sens à cet ouvrage. Même si vous serez peu nombreux à le lire de A à Z...

Résumé

Dans le cadre de la réalité augmentée mobile, un flux vidéo est envoyé à l'utilisateur au travers d'un canal de communication sans fil. Pour garantir une transmission efficace à faible délai, le débit du flux est contrôlé en adaptant les différents paramètres d'encodage en fonction de la bande passante. Le débit peut être diminué en réduisant sa fréquence et/ou en augmentant le facteur de compression du flux vidéo. Ces modifications de paramètres ont un impact sur le niveau de détail et la fluidité perçue par l'utilisateur et donc sur son appréciation subjective. L'expérience ressentie par l'utilisateur dépend également du contexte. Lors d'un mouvement rapide de la tête, la notion de fluidité est plus importante que lorsque le regard est fixe.

Nous proposons un schéma d'adaptation global qui permet de déterminer comment adapter les paramètres d'encodage pour obtenir la meilleure expérience utilisateur possible en fonction d'un contexte dynamique. Par exemple, lorsque l'utilisateur bouge sa tête rapidement, l'image est plus fortement compressée pour permettre d'augmenter la fréquence d'affichage et ainsi d'obtenir une meilleure perception du mouvement.

Le manque de mesure directe pour l'expérience utilisateur subjective est compensé par la conception d'une métrique objective et d'un modèle générique qui permettent de prédire la qualité d'expérience de l'utilisateur en temps réel.

Une stratégie de contrôle de débit basée sur une approche système est développée afin de gérer plusieurs paramètres d'encodage pour contrôler le débit du flux. L'encodeur est modélisé de manière abstraite sous forme de système linéaire monovariable, où la variation du contenu est traitée comme une perturbation. Un contrôleur stable et performant est développé pour le modèle abstrait de l'encodeur.

Pour pouvoir implanter concrètement ce contrôleur, il est nécessaire de déterminer les combinaisons de paramètres de l'encodeur réel qui correspondent à l'entrée unique du modèle abstrait. Un nouvel algorithme direct détermine cette correspondance en temps réel basé sur une méthode de mapping.

L'adaptation contextuelle proposée permet ensuite d'obtenir la combinaison de paramètres d'encodage qui maximise la qualité d'expérience grâce au modèle de qualité d'expérience. Finalement, le schéma d'adaptation global intègre le contrôle de débit, la méthode de mapping et l'adaptation contextuelle pour une implémentation en temps réel.

Des simulations et des mesures expérimentales illustrent l'approche présentée et le schéma d'adaptation global est validé avec différents scénarios.

Mots clés : réalité augmentée, approche système, schéma d'adaptation, qualité d'expérience, contrôle de débit, encodage vidéo, mapping.

Abstract

In the framework of mobile augmented reality, a video stream is sent to the user with the help of a wireless communication link. To guarantee an efficient transmission, the video stream rate is controlled by adapting the encoding parameters such as to follow a given bandwidth. The rate can be reduced by reducing the frame rate and/or by choosing a higher compression factor for the video stream. These parameter modifications impact both the level of detail and the fluidity perceived by the user, and thus his/her subjective appreciation. The experience perceived by the user also depends on the context. During a rapid head motion, the notion of fluidity is more important than for a fixed head position.

We propose an end-to-end adaptation scheme which enables the encoding of parameters such as to provide the best experience for the user regarding the dynamical context. For example, when the user moves quickly his/her head, the frame is compressed more to increase the frame rate and hence achieve a better perception of the motion.

The lack of direct measurement for the subjective user experience is addressed with the design of objective metrics and a generic model to predict the user quality of experience in real time.

A rate control strategy based on a systems approach is deployed to manage the multiple encoding parameters which control the stream rate. The encoder is modeled in an abstract manner as a single-variable linear system, where the content variation is taken

as a perturbation. A stable and efficient controller is designed for the abstract model of the encoder.

To implement the designed controller, the parameter combinations for the real encoder corresponding to the single input of the abstract model should be determined. A new one-pass algorithm determines this correspondence in real time based on a mapping method.

Then, the proposed contextual adaptation enables to get the encoding parameter combination that maximizes the quality of experience using an appropriate model. Finally, the global adaptation scheme combines the rate control, the mapping method and the contextual adaptation for real-time implementation.

Simulation and experiments illustrate the approach and the global adaptation scheme is validated through different scenarios.

Keywords: augmented reality, system approach, adaptation scheme, quality of experience, rate control, video encoding, mapping.

Table des matières

1	Introduction	1
1.1	Objectifs	5
1.2	Postulat	6
1.3	Contributions	6
1.4	Organisation de la dissertation	7
2	Qualité d'expérience	9
2.1	Etat de l'art	10
2.1.1	Qualité d'un flux multimédia	10
2.1.2	Perception dans les environnements virtuels..	13
2.1.3	Définition de la qualité d'expérience	16
2.2	Modélisation de la QoE	17
2.2.1	Représentation du modèle	18
2.2.2	Sélection des variables du modèle	20
2.3	Modèle générique	25
2.3.1	Construction du modèle	26
2.3.2	Simulation du modèle générique	28
2.4	Identification du modèle	32
2.4.1	Evaluation subjective	33
2.4.2	Hypersurface d'approximation	34
2.4.3	Identification expérimentale	35

2.5	Conclusion	41
3	Modèle pour le contrôle de débit	43
3.1	Etat de l'art	44
3.1.1	Contrôle de débit avec notion de qualité	45
3.1.2	Contrôle de débit basé sur l'automatique	46
3.1.3	Approche générique du contrôle de débit	47
3.2	Approche système	48
3.2.1	Encodeur et transmission	49
3.2.2	Contrôle de la transmission	50
3.2.3	Modèle du système transmission	51
3.3	Modèle du système encodeur	53
3.3.1	Représentation non linéaire	53
3.3.2	Modèle linéaire	56
3.4	Conclusion	58
4	Méthode du mapping	61
4.1	Compression DCT	62
4.1.1	Espace de couleurs et échantillonnage	64
4.1.2	Découpage en blocs	65
4.1.3	DCT	67
4.1.4	Quantification	68
4.1.5	Codage entropique	70
4.1.6	Décompression JPEG	71
4.1.7	Taille de l'image compressée	72
4.2	Fonction réalisable dans l'espace linéaire	72
4.3	Définition du mapping	75
4.3.1	Graphique des quatre quadrants	76
4.4	Mapping Φ_2	80
4.4.1	Analyse dans le ρ -domaine	81
4.4.2	Interprétation du ρ -domaine	82
4.5	Algorithme direct	85
4.5.1	Seuils de quantification	86
4.5.2	Histogramme des occurrences	88
4.5.3	Construction du mapping	89

4.5.4	Validation de l'algorithme direct	90
4.6	Application et résultats	92
4.6.1	Choix des paramètres d'encodage	95
4.6.2	Contrôle de débit	97
4.7	Conclusion	103
5	Adaptation	105
5.1	Etat de l'art	106
5.2	Architecture	108
5.3	Adaptation contextuelle	109
5.3.1	Fonctions réalisables	111
5.3.2	Espace d'encodage augmenté	111
5.3.3	Evaluation de la QoE	116
5.3.4	Optimisation	116
5.4	Schéma d'adaptation global	116
5.4.1	Identification du modèle	116
5.4.2	Estimation du contexte	119
5.4.3	Coupe contextuelle	119
5.4.4	Adaptation de transmission	119
5.4.5	Contrôle de débit	119
5.4.6	Mapping	119
5.4.7	Combinaison optimale	120
5.4.8	Système AR	120
5.5	Application	120
5.5.1	Simulation de l'adaptation contextuelle	120
5.5.2	Mesures expérimentales de l'adaptation	124
5.6	Conclusion	129
6	Conclusion	131
6.1	Contributions	132
6.2	Perspectives	133
6.2.1	Réalité augmentée	134
6.2.2	Expérience utilisateur	134
6.2.3	Encodage de flux vidéo	135
6.2.4	Adaptation	136

A Fonctions splines	137
A.1 Splines monovariabes	138
A.2 B-splines monovariabes	140
A.3 Construction d'une spline monovariabes	141
A.4 Construction d'une base multivariabes	141
A.5 Coupe vers un espace de dimension inférieure	145
A.6 Monotonie	148
B Noeuds et coefficients du modèle générique	151
C Sources des images	155
Références	157
Index	167
Acronymes	169
Symboles	171
Termes anglophones	175
Curriculum Vitae	177

Introduction

La réalité augmentée améliore la perception de l'utilisateur en superposant des informations virtuelles sur le monde réel.

Le monde numérique prend de plus en plus de place dans nos vies. Nous sommes constamment confrontés à des ordinateurs ou autres systèmes informatiques : nous prenons notre billet de train dans un distributeur automatique, nous appelons un collègue avec notre *smartphone* (téléphone intelligent) pour l'informer de notre retard, nous travaillons une grande partie de la journée sur un ordinateur, nous regardons un match de hockey sur une télévision équipée d'un récepteur pour la vidéo à la demande et nous discutons à distance avec nos amis dans un univers virtuel. Les informations générées informatiquement sont partout, la barrière entre le réel et le virtuel devient de plus en plus fine. Nous sommes maintenant dans le siècle de l'informatique omniprésente, caractérisée par la présence d'informations numériques dans une multitude d'objets.

L'être humain possède généralement cinq sens pour percevoir son monde environnant : la vue, l'ouïe, le toucher, le goût et l'odorat. La *perception utilisateur* est définie comme l'acquisition d'informations externes en utilisant les sens. Nous utilisons plusieurs sens pour percevoir un événement ou une scène auxquels nous sommes confrontés. Un chirurgien peut par exemple percevoir la gravité de la blessure d'un patient en observant la plaie ou en palpant son pouls.

La réalité augmentée (AR pour *Augmented Reality*) complète notre perception du monde réel en y superposant des informations virtuelles générées par ordinateur (Fig. 1). L'AR combine les mondes réel et virtuel de manière interactive et cohérente. Dans le domaine de la médecine, l'AR permet au chirurgien de voir virtuellement à l'intérieur du patient grâce aux modèles 3D des organes générés à partir de l'imagerie médicale. L'emplacement des tissus atteints par une tumeur peut être mis en évidence. Grâce à l'AR, le chirurgien a accès à des informations supplémentaires qu'il peut utiliser directement pour appuyer ses décisions et améliorer la précision de ses actes. Dans le domaine industriel, les opérateurs ont accès à toutes les informations d'une installation, même à celles qui ne seraient pas visibles directement. Des mesures de température, pression ou niveau de produit sont intégrées virtuellement à l'emplacement adéquat grâce aux images d'AR superposées à l'installation réelle. D'une manière générale, l'AR permet à l'utilisateur de percevoir des informations qui ne seraient pas accessibles autrement. Avec l'AR, un site culturel devient virtuellement actif. Imaginez que vous observez la ville de Berlin à partir de la *Fernsehturm* haute de 200 m. Vous pourriez voir une version virtuelle du Mur de Berlin à son emplacement historique, comme s'il était encore là ; ne vous en souviendriez-vous pas mieux qu'avec les explications du guide audio actuel ?

Supposons un équipement qui permette de superposer des images d'AR à notre vision. Depuis la *Fernsehturm*, le réalisme des objets virtuels doit être suffisant pour pouvoir reconnaître les miradors et différencier les versions de construction du mur. Pour offrir une bonne expérience à l'utilisateur, les graphismes d'AR doivent sembler faire partie de la réalité en tout temps. C'est-à-dire que le niveau de détail et la fréquence d'affichage doivent être élevés pour obtenir une cohérence entre les mondes réel et virtuel. Le modèle virtuel du Mur de Berlin doit rester positionné au bon endroit pour l'utilisateur, même en cas de mouvement rapide de la tête ; le problème d'alignement entre la réalité et les objets virtuels – connu sous le nom



(a) La réalité perçue par l'utilisateur.

(b) Les objets virtuels générés par l'ordinateur.



(c) Scène d'AR vue par l'utilisateur.

Fig. 1: L'utilisateur voit les objets virtuels superposés à la réalité grâce à un dispositif d'affichage.

d'erreur d'alignement¹ – doit être aussi petite que possible pour obtenir une impression naturelle. Le point de vue de l'utilisateur peut être situé à différents endroits, en fonction de sa position et de ses

1. *registration*



Fig. 2: Les graphismes d'AR sont générés puis encodés sous forme de flux vidéo par l'unité de calcul. Le flux vidéo est ensuite envoyé à travers le canal de communication vers l'utilisateur puis affiché sur des lunettes semi-transparentes dans cet exemple.

déplacements. Les graphismes de AR doivent être générés rapidement pour chacun des points de vue pour garder un bon alignement.

Lorsque l'utilisateur se déplace sur la terrasse de la tour pour observer chaque partie du Mur de Berlin virtuel intégré dans la ville, l'équipement qu'il porte sur lui pour percevoir l'AR doit être mobile. Une solution est de limiter au minimum l'équipement porté par l'utilisateur et de déporter les unités de calcul au centre de la tour pour générer les graphismes d'AR. Ces graphismes sont encodés sous forme de flux vidéo, puis envoyés à travers un canal de communication sans fil. Finalement, l'équipement mobile de l'utilisateur reçoit, décode puis affiche ce flux vidéo dans le visiocasque (HMD pour *Head Mounted Display*) de l'utilisateur (Fig. 2). Pour garantir une transmission efficace, le flux vidéo doit être compressé en fonction de la bande passante disponible du canal de communication. Un mécanisme de contrôle de débit² doit adapter les paramètres d'encodage pour réguler le débit³ du flux vidéo.

La réduction du débit du flux vidéo est obtenue en diminuant la fréquence des images ou la taille (en bits) des images compressées. La

2. *rate control*

3. *rate*

fluidité d'une scène en mouvement est liée à la vitesse du mouvement et à la fréquence des images. De même, son réalisme dépend du niveau de détail de l'image qui est généralement lié au volume de données représentant l'image et donc sa taille.

L'*expérience utilisateur* représente l'appréciation subjective perçue par un utilisateur qui interagit avec une interface. L'expérience utilisateur est l'élément qui doit donner envie à l'utilisateur de continuer à interagir avec une interface. Des changements dans la fluidité ou le réalisme d'une scène d'AR ont un impact sur l'expérience utilisateur. D'autres éléments ont une influence sur l'expérience utilisateur pour l'AR. Les caractéristiques de l'utilisateur, comme sa sensibilité, son acuité visuelle ou sa capacité à appréhender un tel système peuvent changer son niveau de satisfaction. Des éléments extérieurs liés au *contexte* d'utilisation peuvent également influencer l'expérience utilisateur. Par exemple en présence d'un danger, l'utilisateur aura une perception différente et son comportement va changer ; ses mouvements seront plus rapides et il traitera en priorité les informations qui pourraient l'aider à résoudre la situation.

Le problème exposé peut se résumer à la question suivante :
Comment adapter les paramètres d'encodage pour obtenir la meilleure expérience utilisateur possible en fonction d'un contexte dynamique donné ?

1.1 Objectifs

Cette dissertation s'articule autour de l'adaptation de l'encodage dans le domaine de l'AR, en tenant compte du contexte et de la perception de l'utilisateur. L'objectif principal est de trouver une méthode qui garantit la meilleure expérience utilisateur possible, sans toucher à la conception ou à l'architecture de l'application d'AR. Pour cela, il faut définir la manière d'adapter automatiquement les paramètres d'encodage en fonction du contexte et développer un contrôle de débit qui prend en compte les spécificités de l'encodage vidéo de l'AR. Ces deux concepts sont intégrés dans une approche globale.

1.2 Postulat

Dans cette dissertation, nous montrons qu'une approche système alliée au principe de rétroaction permet de développer de manière efficace des méthodes pour améliorer l'expérience utilisateur dans le domaine de l'AR. La représentation abstraite des systèmes permet de développer un schéma d'adaptation global. Les différents systèmes sont modélisés au niveau d'abstraction approprié. Cela permet de synthétiser les algorithmes d'adaptation, tant au niveau de l'expérience utilisateur que pour le contrôle de débit. La gestion simultanée de plusieurs paramètres pour l'encodage du flux vidéo de l'AR permet d'adapter l'encodage en fonction du contexte afin d'atteindre une meilleure expérience utilisateur.

1.3 Contributions

Dans cette dissertation, nous présentons une méthodologie pour adapter les paramètres d'encodage. Les contributions apportées par cette thèse sont les suivantes :

1. **Conception d'un modèle générique pour la qualité d'expérience.** Ce modèle objectif permet de prédire la qualité d'expérience d'un utilisateur en temps réel en n'utilisant que des mesures liées au contexte et à l'encodage.
2. **Représentation du modèle de qualité d'expérience par une fonction mathématique.** Le modèle est défini par une fonction spline multivariable. Le modèle peut être utilisé pour faire de l'adaptation grâce à la dissociation des variables manipulées et des variables mesurées.
3. **Stratégie de contrôle de débit basée sur une approche système.** Un modèle générique linéaire d'encodeur permet de développer un contrôleur et de prendre en compte de façon abstraite plusieurs paramètres d'encodage. La stabilité et la performance du contrôle de débit peuvent ainsi être analysées.

4. **Algorithme direct de correspondance entre le modèle générique et un encodeur.** Cet algorithme permet de déterminer les valeurs des paramètres réels de l'encodeur à partir de la variable de sortie du contrôle de débit. Ce *mapping* (mise en correspondance) est calculé en temps réel pour chaque image.
5. **Schéma d'adaptation global.** Les paramètres d'encodage sont adaptés en temps réel pour maximiser la qualité d'expérience. Malgré les contraintes de bande passante ou les événements contextuels, l'utilisateur obtient la meilleure qualité possible d'AR.

1.4 Organisation de la dissertation

Dans le Chap. 2, nous étudions l'expérience utilisateur dans le domaine de l'AR et identifions les éléments ayant un impact sur elle. Certains de ces éléments sont liés au contexte et d'autres à l'encodage vidéo. Nous définissons une métrique objective nommée qualité d'expérience (QoE pour *Quality of Experience*) qui est ensuite utilisée pour nos analyses et synthèses. Nous construisons ensuite un modèle de QoE qui représente l'impact des éléments contextuels et des paramètres d'encodage sur la QoE.

Lorsqu'on prend en compte un encodeur évolué basé sur la transformée en cosinus discrète (DCT pour *Discrete Cosine Transform*), il est nécessaire d'implémenter un contrôle de débit afin de maîtriser le débit du flux vidéo. En effet, pour des paramètres d'encodage fixe, le débit varie en fonction du contenu du flux vidéo. Dans le Chap. 3, nous proposons une approche système basée sur un modèle linéaire qui permet d'exploiter les outils de l'automatique permettant d'analyser les caractéristiques dynamiques. Ainsi, le contrôleur est synthétisé et intégré dans une boucle de rétroaction.

Le modèle de l'encodeur n'étant pas linéaire, il est judicieux de le transformer en une représentation linéaire. Le Chap. 4 propose un changement de variable pour obtenir la représentation désirée. Le

passage entre les deux représentations se fait à l'aide d'un *mapping*, qui est différent pour chaque image. Ce *mapping* est nécessaire pour appliquer le contrôle de débit à l'encodeur réel. Nous proposons une identification en temps réel de ce *mapping* qui permet de mettre en oeuvre le contrôle de débit en évitant d'augmenter le délai.

En utilisant une approche système et le principe de rétroaction, nous développons une stratégie d'adaptation qui permet d'obtenir la meilleure QoE possible en temps réel (Chap. 5). En fonction de la bande passante disponible, les différentes combinaisons des paramètres d'encodage sont déterminées. A chaque instant, les informations contextuelles sont mesurées et la partie du modèle correspondant à ce contexte actuel est utilisée pour choisir la meilleure combinaison des paramètres d'encodage. Enfin, l'adaptation contextuelle et le contrôle de débit sont intégrés dans un schéma d'adaptation global qui permet de garantir la meilleure QoE possible en fonction du contexte, et ce, en dépit de la variation du contenu du flux vidéo.

Le Chap. 6 conclut cette dissertation. Les principales contributions y sont rappelées et quelques pistes d'amélioration y sont proposées.

Qualité d'expérience

L'étude de la qualité d'expérience permet de développer de nouvelles solutions pour améliorer l'expérience de l'utilisateur.

De nombreuses métaphores d'interaction homme-machine telle que l'AR seraient technologiquement prêtes à être utilisées à large échelle. Cependant, nombre d'entre elles ont du mal à s'imposer car elles ne répondent pas aux attentes des utilisateurs. L'*expérience utilisateur* est donc un point important auquel il faut tenir compte pour le développement de nouvelles technologies.

Dans ce chapitre, nous étudions l'expérience utilisateur dans le contexte de l'AR mobile, où les données d'AR sont encodées sous forme de flux vidéo. Nous proposons un modèle pour représenter l'expérience utilisateur. Ce modèle permet de prédire la qualité d'expérience (QoE pour *Quality of Experience*) de l'utilisateur et ainsi de déterminer comment offrir à l'utilisateur la meilleure QoE possible, sans avoir à questionner continuellement l'utilisateur. La meilleure QoE possible est obtenue pour un contexte courant de l'utilisateur, en fonction des contraintes de l'application d'AR qui ne peuvent pas être dépassées. Par exemple, il n'est pas possible de d'augmenter le réalisme au delà des capacités matérielles et logicielles.

Le modèle de QoE permet d'obtenir une métrique objective pour la QoE, calculée à partir du modèle et des valeurs de ses variables. Le modèle peut être construit arbitrairement ou sur la base d'éva-

luations subjectives. Le modèle de QoE sera ensuite utilisé dans le schéma d'adaptation présenté dans le Chap. 5 pour adapter les paramètres d'encodage en fonction du contexte.

La Sect. 2.1 présente l'état de l'art lié à l'expérience utilisateur dans le domaine de l'AR. Nous y définissons également la QoE. Dans la Sect. 2.2, la forme du modèle de QoE est proposée et les variables du modèle sont sélectionnées. La Sect. 2.3 présente un modèle générique de QoE basé sur une fonction spline multivariable, puis son utilisation est illustrée lors des simulations. La démarche pour identifier le modèle de QoE avec des évaluations subjectives est présentée dans la Sect. 2.4. Finalement, la Sect. 2.5 rappelle les contributions de ce chapitre et présente les perspectives.

2.1 Etat de l'art

De nombreuses études ont été faites dans le domaine de l'expérience utilisateur, sans toutefois proposer un modèle de qualité pour l'AR. Le flux AR est un flux multimédia avec des caractéristiques spécifiques liées à la perception et notamment concernant le délai. Nous présentons dans les prochains paragraphes les travaux significatifs concernant la modélisation de la qualité des flux multimédias et la perception dans les environnements virtuels. Ceci correspondant au domaine englobant autant l'AR, la réalité virtuelle (VR pour *Virtual Reality*) et la réalité mixte.

2.1.1 Qualité d'un flux multimédia

L'évaluation de la qualité vidéo prend une place considérable dans les études liées à l'expérience utilisateur. L'impact de la variation de paramètres d'encodage sur la perception de l'utilisateur est quantifié dans le but de déterminer un moyen adéquat pour encoder la vidéo en fonction des contraintes de transmission de données et de la variation du contenu. Engelke et Zepernick ont présenté une étude générale sur les métriques de qualité de la vidéo basés sur la

perception [1]. Les mesures objectives, comme le rapport crête signal sur bruit (PSNR pour *Peak Signal to Noise Ratio*), sont souvent utilisées mais sont faiblement corrélées avec la qualité réellement perçue par l'utilisateur. Le concept de QoE a été introduit pour y remédier. Les auteurs distinguent les méthodes d'évaluation subjective, qui nécessitent des évaluations laborieuses et des méthodes objectives perceptuelles, qui prédisent la qualité de manière automatique. L'approche psychophysique intègre des aspects du système visuel humain (HVS pour *Human Visual System*) dans la modélisation alors que l'approche ingénieur est principalement basée sur l'analyse de l'image et de ses caractéristiques.

De nombreux modèles de la perception humaine et des métriques liées à la qualité ont été développés pour prédire la qualité perçue par l'utilisateur et améliorer la performance des systèmes d'encodage ou de transmission. Les approches diffèrent en fonction de la nature du flux multimédia. Ghinea et son équipe se sont intéressés à la qualité de perception (QoP pour *Quality of Perception*) pour les présentations multimédia à contenu informatif, encodées et transmises à travers un réseau de communication [2–6]. Dans leurs études, les auteurs différencient la satisfaction liée à la qualité du média, de la capacité à analyser l'information contenue dans le média. Certains paramètres d'encodage comme l'augmentation de la fréquence d'affichage ou du délai de transmission ont un effet sur la satisfaction mais pas forcément sur l'assimilation des informations. En tenant compte de l'assimilation de l'information, Chen et al. ont montré l'importance de s'adapter à l'utilisateur en étudiant la QoP en fonction du style cognitif de l'utilisateur [7], où certains préfèrent recevoir l'information à travers une image alors que d'autres préfèrent lire ou entendre des mots. Dans le domaine de la vidéo, Pereira propose de caractériser la QoE avec un modèle comprenant des notions sensorielles, perceptuelles et émotionnelles [8]. Perkis et al. ont présenté un modèle en forme d'arbre intégrant des branches pour les paramètres objectifs et subjectifs afin de mesurer la QoE des services multimédia [9].

L'approche psychophysique qui prend en compte les connaissances du HVS est régulièrement utilisée pour construire des modèles ou des métriques [10,11]. Les mécanismes pour la perception des couleurs, des formes et des mouvements ont été utilisés par Winkler pour construire la métrique de distorsion perceptuelle (PDM pour *Perceptual Distortion Metric*) permettant d'évaluer la qualité vidéo [12].

Des métriques de qualité plus spécifiques au domaine des communications ont été développées pour remplacer avantageusement les mesures objectives, en tenant compte des caractéristiques de la transmission [13–15]. La diminution du débit de la vidéo ou l'augmentation des pertes de paquets engendrent une baisse dans la qualité de l'image. Cranley et al. proposent une manière optimale d'encoder une vidéo sujette au *streaming* (diffusion de flux continu) pour un débit donné [16]. Cependant, les mécanismes de *streaming* engendrent un délais qui n'est pas acceptable pour notre domaine d'application d'AR. Les auteurs proposent d'utiliser la trajectoire d'adaptation optimale (OAT pour *Optimal Adaptation Trajectory*) dans l'espace d'adaptation pour représenter la configuration d'encodage qui correspond le mieux à la complexité spatiale et temporelle de la vidéo. L'OAT est déterminée grâce à des évaluations subjectives entreprises précédemment. D'autres études montrent que le contenu spatio-temporel d'une vidéo a un impact sur la qualité perçue par l'utilisateur [17,18]. De ce fait, il est judicieux d'adapter la fréquence d'affichage ou le débit de transmission en fonction de la dynamique plutôt que de le laisser fixe.

Ces techniques font référence à des flux multimédias proches du domaine de l'AR, mais ne peuvent pas s'y appliquer directement. En effet, des notions différentes de perception interviennent lorsqu'un utilisateur est confronté à un environnement virtuel. La tête et le regard de l'utilisateur restent principalement statiques lors du visionnement d'une vidéo, alors qu'ils sont généralement en mouvement dans les environnements virtuels. L'AR se distingue également du visionnement de vidéo par l'alignement¹ entre les éléments vir-

1. *registration*

tuels et le monde réel. Les problèmes d'alignement sont principalement dus au délai. Comme cela joue un rôle prépondérant pour l'expérience utilisateur dans l'AR, les études présentées ci-dessus ne sont pas adaptées à notre problématique.

2.1.2 Perception dans les environnements virtuels

La perception et l'expérience utilisateur ont également été étudiées dans le domaine des environnements virtuels. Nous proposons un modèle qui prend en compte plusieurs éléments qui ont un impact sur l'expérience utilisateur, alors qu'ils sont généralement étudiés séparément.

Azuma explique que les erreurs d'alignement apparaissent à cause du délai global² [19]. Le délai global du système est défini comme la différence entre le moment où le système de positionnement³ mesure la *pose* (combinaison de position et orientation) du point de vue, et le moment où les images générées par rapport à ce point de vue sont affichées. Un délai global de 100 ms est typique dans les systèmes d'AR existants. Certains systèmes plus simples peuvent obtenir un délai plus faible, mais d'autres ont un délai plus long notamment dans le cas de transmission sur un canal de communication chargé. Ce délai peut causer de grandes erreurs d'alignement, ce qui perturbe l'illusion de coexistence entre les mondes réel et virtuel. A titre d'exemple, avec un délai de 100 ms et une vitesse de rotation de la tête de 50 degrés par seconde, l'erreur angulaire dynamique est de 5 degrés. A bout de bras, cela correspond à une erreur d'alignement de presque 60 mm.

Pasman et Jansen ont évalué que l'utilisateur accepte un délai autour de 10 ms et ont développé un système d'AR mobile avec un faible délai [20].

2. *end-to-end*

3. *tracking*

La sensibilité du délai durant un mouvement de tête dans un environnement virtuel a été identifiée la première fois par Ellis et al [21]. Les utilisateurs doivent différencier des scénarios utilisant des délais différents. Lors d'un mouvement de la tête, ils ne sont pas capables de détecter de différences lorsque l'écart entre les délais est inférieur au seuil de 30 ms. Allison et al. ont complété cette étude en tenant compte de la vitesse moyenne de la tête [22]. Un utilisateur aurait un seuil de perception entre 60 ms et 200 ms pour des vitesses angulaires décroissant de 90 à 22.5 degrés par seconde. D'autres études ont montré que l'utilisateur n'était pas capable de discriminer des délais différents de moins de 17 ms, que ce soit en terme de mouvement relatif entre les mondes réel et virtuel dans un environnement virtuel ou pour des scènes de plus grande complexité [23,24]. Plus récemment, constatant que la perception du mouvement relatif n'est pas encore bien comprise quand la tête est elle-même en mouvement, Jerald et al. ont remarqué que la sensibilité au mouvement est plus haute vers la fin du mouvement alors qu'elle n'est pas si importante durant toute la durée du mouvement [25]. Comme les méthodes et les résultats de ces études diffèrent, un consensus à ce sujet n'a pas encore été trouvé.

La fréquence d'affichage a également un effet sur la QoE. La sensation d'immersion augmente quand la fréquence augmente. Un niveau d'interaction réaliste est obtenu avec une fréquence de 20 Hz [26]. Des tests subjectifs ont montré que les performances pour juger la variation de dimension d'un objet virtuel sont significativement plus faibles quand la fréquence diminue de 20 Hz à 10 Hz [27].

Perritaz et al. ont étudié l'effet couplé de la fréquence et de la compression MPEG (Moving Picture Experts Group) sur l'expérience de l'utilisateur équipé d'un visiocasque (HMD pour *Head Mounted Display*) dans lequel l'utilisateur voyait un flux vidéo compressé de la réalité [28]. Les résultats des tests subjectifs sont similaires à ceux obtenus par Cranley et al. [16] alors que les conditions n'étaient pas comparables. Dans l'étude de Cranley et al., les séquences vidéo étaient projetées sur un écran d'ordinateur sans aucune interaction.

La résolution de l'image a aussi un effet sur la performance et la QoE. Des expériences ont montré qu'une diminution de la résolution entraîne une baisse du sentiment d'immersion et de plus faibles performances même pour de simples tâches [29]. Ryu et al. ont même montré que la résolution de l'image influence l'estimation de la distance dans un environnement virtuel où les images sont projetées sur un grand écran [30]. Lors des tests, les sujets avaient tendance à sous-estimer la distance dans l'espace virtuel pour des résolutions basses. La notion de distance a également été étudiée dans le cadre de l'AR stéréoscopique, où les facteurs importants, tels que la taille relative des objets, l'angle de vue ou la résolution d'affichage, ont été classifiés [31].

Whalen et al. ont décrit une méthode pour évaluer la QoE dans les environnements virtuels en intégrant les mesures physiologiques de stress et d'activité cérébrale [32]. Ces différentes mesures peuvent être utilisées pour qualifier la notion de contexte. Le contexte représente les conditions d'utilisation qui influencent l'expérience utilisateur. Ainsi, une situation d'urgence de l'utilisateur pourrait être remarquée grâce à une augmentation de sa fréquence cardiaque.

En résumé de ces différentes études, il ressort que les éléments qui ont un impact important sur l'expérience utilisateur dans le domaine de l'AR sont :

- l'alignement statique et dynamique (dû au délai)
- le niveau de détail de l'image
- la vitesse de déplacement de la tête
- l'interaction avec l'environnement.

Tout comme pour les flux multimédia, on retrouve différentes méthodes spécifiques à l'objectif de la recherche pour évaluer la qualité d'expérience de l'utilisateur, mais pas de modèle générique d'expérience utilisateur pour l'AR.

2.1.3 Définition de la qualité d'expérience

Dans la littérature, différents termes font référence à l'expérience utilisateur et plus précisément à la QoE. Les standards pour les transmissions d'ITU (International Telecommunication Union) définissent la QoE comme étant l'acceptabilité globale d'une application ou d'un service, telle qu'elle est perçue subjectivement par l'utilisateur final [33]. La QoE comprend tous les effets du système global, dont la perception peut être influencée par les attentes de l'utilisateur et le contexte. Une notion plus générale de la QoE est utilisée par Alben pour définir tous les aspects d'interaction entre une personne et un produit [34]. Jain présente la QoE comme étant un élément important à étudier et utiliser dans le domaine du multimédia [35]. Comparée à la qualité de service (QoS pour *Quality of Service*), la QoE comprend des notions de subjectivité et de dépendance au contexte. Des mesures sont nécessaires pour comprendre la QoE et développer des systèmes qui s'adaptent aux besoins des utilisateurs. McNamara et Kirakowski ont développé le concept d'utilisabilité en différenciant la qualité d'utilisation (QoU pour *Quality of Use*) de la QoE [36]. Alors que la QoU ne traite que l'interaction objective entre l'utilisateur et le produit, les aspects subjectifs sont représentés dans la notion de QoE. L'étude de la QoE peut être approchée de trois manières différentes : ajouter une nouvelle dimension à la structure de l'utilisabilité, proposer un nouveau concept distinct de l'utilisabilité ou regarder le concept d'expérience avec un regard plus théorique et philosophique. Les auteurs rappellent que les méthodes et l'interprétation des données sont propres à chaque domaine d'investigation.

La littérature propose un grand nombre de définitions pour la QoE. Notre définition permet d'utiliser une approche système dans le but de mettre en place un schéma d'adaptation.

Définition 1 (Qualité d'expérience). *La QoE est la métrique qui représente la satisfaction subjective de l'utilisateur, relative à l'exploitation d'un système d'AR, dans un but donné et un contexte donné. Elle est définie entre 0 (mauvaise) et 1 (optimale).*

2.2 Modélisation de la QoE

L'évaluation de la qualité peut se faire de manière subjective ou de manière objective. Il existe une combinaison de ces méthodes, comme l'évaluation objective perceptuelle.

L'*évaluation subjective* se base sur les notes d'appréciation d'un groupe d'utilisateurs auquel différentes configurations sont présentées. Les résultats sont analysés pour déterminer la configuration offrant la meilleure qualité ou quantifier l'effet de la modification dans les configurations. Ces méthodes d'évaluation subjective rendent bien l'impression finale de l'utilisateur pour un environnement donné. Cependant, elles requièrent du temps pour être mises en oeuvre et ne peuvent donc pas être utilisées pour l'évaluation de la qualité en temps réel.

L'*évaluation objective* utilise des métriques objectives liées au rendu final. Dans le cadre de l'évaluation de la qualité d'image, la métrique traditionnelle est le PSNR qui correspond au rapport entre le signal transmis de l'image originale et le bruit dû aux artefacts d'encodage. Cette métrique nécessite l'image de référence, ce qui n'est pas toujours possible, la classant ainsi dans les méthodes basées sur une référence⁴. D'autres méthodes ne nécessitant pas ou qu'une partie de la référence ont également été développées pour l'évaluation de la qualité d'image [1]. Les méthodes d'évaluation objective ont l'avantage de pouvoir être utilisées pour l'estimation de la qualité en temps réel. Cependant, elles sont fortement critiquées pour leur manque de pertinence et de corrélation par rapport aux méthodes subjectives [10].

L'*évaluation objective perceptuelle* de qualité regroupe les avantages des évaluations subjective et objective [37]. La pertinence de l'évaluation subjective est alliée à la capacité d'estimer la qualité en temps réel comme pour l'évaluation objective. Les méthodes objectives perceptuelles se basent généralement sur un modèle qui émule

4. *reference-based*

le canal de perception humain, construit sur la base de la connaissance de la perception humaine.

Dans cette dissertation, nous proposons un *modèle objectif perceptuel* qui permet d'estimer la QoE sur la base de la valeur des variables prises en compte. Les variables correspondent aux éléments quantifiables qui ont un impact sur l'expérience utilisateur. Les sections suivantes présentent les différentes variables prises en compte dans le modèle ainsi que la représentation mathématique du modèle. La méthode développée peut être utilisée dans divers domaines, mais nous présentons la démarche pour obtenir un modèle de QoE pour l'AR.

2.2.1 Représentation du modèle

Dans le cadre de l'AR, nous avons classifié les variables qui ont un impact sur l'expérience utilisateur en trois groupes principaux :

- les variables de conception
- les paramètres d'encodage
- les variables contextuelles.

Les *variables de conception* correspondent aux aspects liés au matériel et à l'ergonomie. La dimension de l'écran, le poids du système d'AR, l'infrastructure de communication et l'erreur statique de positionnement font partie de ce premier groupe. Ces variables sont déterminées lors de la conception et sont généralement fixes.

Les *paramètres d'encodage* sont liés au réalisme de la scène virtuelle ou de sa représentation en tant qu'image. Ces paramètres correspondent par exemple au nombre de polygones composant un objet virtuel, la profondeur de couleurs, la résolution de l'image affichée ou l'intervalle de temps entre deux images successives.

Les *variables contextuelles* ont également un impact important sur l'expérience utilisateur. Dans une situation de stress, par exemple lors d'une alarme, l'utilisateur ne va pas apprécier l'expérience d'AR de la même manière que durant son utilisation habituelle. L'AR est également appréhendée différemment en fonction des connaissances

préalables ou du niveau d'expertise de son utilisateur. D'autres variables contextuelles comme la santé ou l'acuité visuelle de l'utilisateur, sa vitesse de déplacement ou le degré de difficulté de la tâche modifient également l'expérience utilisateur. Certaines de ces variables peuvent être mesurées et leur impact sur l'expérience utilisateur peut être quantifié (Sect. 2.2.2).

Les paramètres de conception ne changent pas au cours du temps et ne sont donc pas pris en compte pour notre étude d'adaptation en temps réel. Nous considérons dès lors uniquement les paramètres d'encodage et les variables contextuelles car leur évolution dynamique modifie l'expérience utilisateur en temps réel.

Définition 2 (Modèle de qualité d'expérience). *Le modèle de QoE représente l'impact des paramètres d'encodage v_e et des variables contextuelles v_c sur la métrique QoE. Ce modèle est exprimé sous la forme d'une fonction M tel que :*

$$QoE = M(v_e, v_c). \quad (1)$$

Une *hypersurface* est choisie pour représenter le modèle qui lie les variables sélectionnées à la QoE. L'hypersurface correspond à la généralisation du concept de surface pour des dimensions supérieures à trois. La Sec. 2.3 présente le choix d'une fonction lisse pour représenter cette surface.

Dans le Chap. 5, le modèle de QoE est utilisé pour choisir les valeurs des paramètres d'encodage en temps réel. Les paramètres d'encodage y seront traités comme des variables manipulées alors que les variables contextuelles ne seront que mesurées. Afin de pouvoir mettre en oeuvre le schéma d'adaptation du Chap. 5, nous proposons de représenter l'hypersurface dans un espace qui ne tient plus compte des variables contextuelles.

Définition 3 (Espace d'encodage). *L'espace d'encodage est l'espace engendré par les paramètres d'encodage v_e . La dimension de cet espace correspond au nombre de paramètres d'encodage pris en compte. L'espace d'encodage augmenté correspond à l'espace d'encodage avec la métrique QoE comme dimension supplémentaire.*

Définition 4 (Modèle contextuel). *Le modèle contextuel de QOE $M_c(v_e, k)$ correspond à la section de coupe du modèle $M(v_e, v_c)$ dans l'espace d'encodage augmenté à un instant discret k . A cet instant k correspond un contexte donné, représenté par les valeurs des variables contextuelles $v_c(k)$.*

$$QoE = M_c(v_e, k) \quad (2)$$

Le modèle contextuel dépend de l'instant k , mais plus du tout des variables contextuelles v_c . Des hypersurfaces de dimension inférieure au modèle M représentent des contextes différents dans l'espace d'encodage augmenté.

Comme l'utilisateur perçoit l'expérience de manière continue, nous faisons l'hypothèse que la valeur de la métrique QOE de l'utilisateur subit un maintien d'ordre zéro entre les instants discrets; un instant discret k correspondant à la génération d'une nouvelle image d'AR.

2.2.2 Sélection des variables du modèle

Les études sur la perception dans les environnements virtuels ont montré les différentes variables qui influencent l'expérience utilisateur (Sect. 2.1.2). Dans le cadre de cette thèse, nous retenons :

- la fréquence de génération des images f
- le facteur de *quantification*⁵ de l'image q
- le délai global d
- la vitesse angulaire de la tête s .

Les deux premières variables f et q font partie des paramètres d'encodage, alors que les deux dernières d et s sont des variables contextuelles. Ces quatre variables sont utilisées pour le schéma d'adaptation présenté au Chap. 5.

5. *quantization*

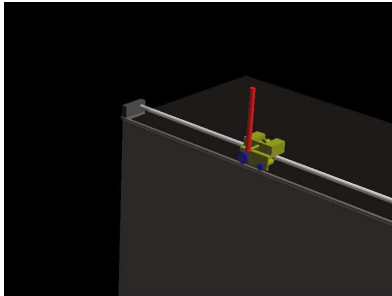
La Fig. 3 illustre le délai et la quantification de l'image, où un objet virtuel est placé dans un environnement réel. L'environnement réel est représenté par une table émulée pour représenter la vision de transparence de l'utilisateur, alors qu'un pendule inversé monté sur son rail fait office d'objet virtuel (Fig. 3a). La Fig. 3b illustre l'erreur d'alignement dynamique due au délai global pour une vitesse relative entre les mondes réel et virtuel, par exemple quand l'utilisateur tourne la tête. La Fig. 3c montre la scène virtuelle quantifiée avec une densité de pixels plus faible.

Paramètres d'encodage

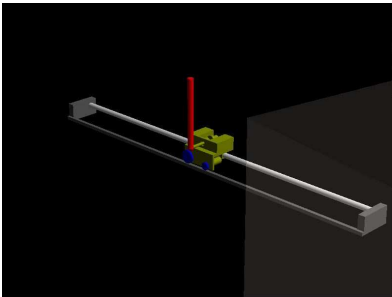
Les paramètres d'encodage (fréquence et quantification) influencent le débit du flux vidéo, mais également la qualité du flux vidéo et donc l'expérience utilisateur.

Le *facteur de quantification de l'image* modifie la qualité intrinsèque de l'image. Nous utilisons ici la notion de facteur de quantification de façon générique, sans se restreindre à une transformation spécifique. La résolution en pixels de l'image, le facteur de compression spatial, la profondeur de couleurs ou le niveau de détail des objets virtuels sont des facteurs de quantification. Avec une valeur élevée du facteur quantification (par exemple une résolution élevée), l'image quantifiée reste très proche de l'image originale et donc la qualité reste élevée. De même, un faible facteur de quantification (par exemple un petit nombre de couleurs) engendre une image de qualité plus basse. Un facteur de quantification spécifique est choisi dans certaines sections de cette dissertation en fonction de l'objectif choisi ou des contraintes matérielles et logicielles.

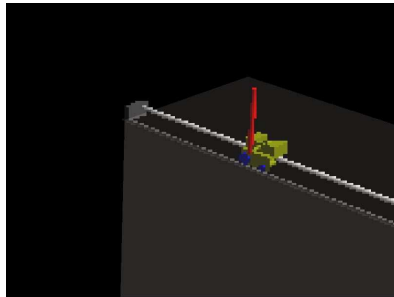
La *fréquence de génération des images*, correspond à la fréquence à laquelle les images sont produites. Cette fréquence modifie l'expérience utilisateur. Dans le cas d'une scène statique avec un point de vue en mouvement, l'objet virtuel reste parfaitement aligné à l'environnement réel uniquement quand la nouvelle image vient d'être affichée. Plus le laps de temps entre l'affichage de deux images est



(a) L'objet virtuel est placé sur une table réelle. La table est émulée avec un parallélépipède gris idéal. Elle est affichée sans délai, avec une fréquence et une résolution maximales.



(b) Illustration de l'erreur dynamique d'alignement pour un délai de 100 ms et une vitesse relative de 20 degrés par seconde.



(c) La scène virtuelle est générée avec une densité de 179×134 pixels, puis est projetée à la résolution d'affichage de 800×600 . La table reste générée avec une densité maximale de 800×600 .

Fig. 3: Illustration de l'effet du délai global et de la quantification (densité de pixels) de l'image sur l'expérience utilisateur.

grand, plus l'erreur d'alignement est grand. Ce délai s'ajoute au délai global. Il n'est pas perceptible quand la fréquence est élevée, mais peut devenir gênant quand la fréquence est basse.

Les contraintes d'implémentation ne permettent pas nécessairement d'avoir une fréquence et un facteur de quantification élevés. La puissance de calcul limitée ou une bande passante étroite du canal de communication peuvent en être la cause, d'où la nécessité d'adapter ces paramètres. Il est possible de contrôler la fréquence en modifiant l'intervalle de temps entre deux rendus successifs de la scène virtuelle. De même, chaque scène peut être encodée sous forme d'images ayant une résolution ou des paramètres de compression différents.

Plus la fréquence ou le facteur de quantification est faible, plus la QoE est faible. Il y a donc une relation *monotone* entre les paramètres de quantification et la QoE. Le modèle doit contenir cette notion de monotonie.

Délai global

Le délai global est une variable ayant un impact important sur l'expérience utilisateur. Prenons le cas d'un système d'AR où les mesures de positionnement et l'alignement mécanique des éléments d'affichage sont parfaits; en d'autres termes, il n'y a pas d'erreur statique d'alignement. Lors d'un changement de point de vue de l'utilisateur, si le délai est nul, les mondes réel et virtuel resteront parfaitement alignés. Par contre si le délai n'est pas nul, les objets virtuels sembleront traîner derrière leur position supposée dans le monde réel. Plus le délai est élevé, plus la cohérence entre les mondes réel et virtuel est faible. Le délai est classifié soit comme une variable de conception s'il reste fixe, soit comme une variable contextuelle s'il varie.

L'écart entre la réalité et la position des objets virtuels est appelé erreur dynamique d'alignement. Pour l'AR mobile, le délai global est la somme des délais du positionnement, de la génération de la scène virtuelle, de l'encodage, de la transmission, du décodage et de

l'affichage. Certains d'entre eux sont fixes alors que d'autres sont variables et peuvent être contrôlés. Par exemple, un changement de complexité de la scène virtuelle peut influencer le délai de la génération de la scène et celui des procédés d'encodage. Certains de ces délais peuvent être mesurés ou approximés. Il existe par exemple des méthodes pour mesurer le délai global dans une application d'AR avec transparence vidéo [38].

De manière similaire aux paramètres d'encodage, la QoE diminue quand le délai global augmente. Il y a donc aussi une relation monotone entre le délai et la QoE à intégrer dans le modèle.

Mouvement de la tête et des yeux

Comme expliqué dans la section précédente, la combinaison du délai et d'un changement de point de vue de l'utilisateur engendre une erreur dynamique d'alignement. Cette section explique et quantifie cette notion de changement de point de vue.

Une image d'AR provient de la projection d'une scène virtuelle immobile perçue par un point de vue donné, appelé généralement *caméra virtuelle*. Cette image est ensuite affichée pour l'utilisateur. Dans le dispositif de génération d'images, la caméra virtuelle est placée de telle sorte que la scène virtuelle projetée dans l'image soit superposée au monde réel. La *pose* de la caméra virtuelle est calculée en fonction du point de vue de l'utilisateur, de la *pose* du dispositif d'affichage et de la *pose* de la scène virtuelle dans le référentiel réel.

L'erreur dynamique d'alignement apparaît lorsque l'image est affichée avec du délai et que la caméra virtuelle est en mouvement. Plus le mouvement est rapide, plus l'erreur est grande. Il est donc nécessaire de quantifier la vitesse de la caméra virtuelle.

Dans le cas de l'affichage à travers un dispositif solidaire à la tête, par exemple un HMD, il est courant d'utiliser les hypothèses suivantes [39, 40] :

- la vision humaine est modélisée avec un modèle simple de caméra⁶
- la direction du regard est fixe et alignée à la tête
- l'écran d'affichage est centré et perpendiculaire à la direction du regard.

Dans ce cas, la vitesse de la caméra virtuelle correspond à la vitesse de déplacement de la tête. Il existe de nombreuses manières de quantifier la vitesse de la tête. La dynamique de translation de la tête étant beaucoup plus faible que la dynamique de rotation, nous considérons uniquement la vitesse de rotation de la tête. Dans d'autres cas, il peut être nécessaire de quantifier également le mouvement des yeux [41] ou la dynamique de la scène virtuelle.

En résumé, nous considérons la norme de la vitesse de rotation de la tête pour quantifier le changement de point de vue qui engendre l'erreur dynamique d'alignement pour une scène virtuelle immobile.

2.3 Modèle générique

Le modèle de QoE représente l'impact des variables contextuelles et des paramètres d'encodage sur la métrique QoE. Cette section présente la fonction de l'hypersurface qui correspond au modèle ainsi qu'un modèle générique M_g qui sera utilisé pour prédire la QoE. La fonction M_g du modèle QoE = $M_g(v_e, v_c)$ est une hypersurface. Nous avons sélectionné une fonction spline multivariable (Annexe A) pour représenter M_g . Une spline est une fonction qui offre de nombreux avantages, tant au niveau de la flexibilité que de l'utilisation. Cette fonction répond à nos besoins concernant la monotonie, la coupe dans un sous-espace et la possibilité d'approximer des points de mesure.

Cette section présente la construction du modèle générique. Ce modèle est ensuite utilisé pour évaluer l'évolution de la métrique QoE en simulation.

6. *pinhole camera model*

2.3.1 Construction du modèle

Cette section présente la construction d'un modèle de QoE générique. Le modèle est identifié pour un cas particulier dans la Sect. 2.4.

Une spline multivariable monotone est utilisée pour représenter le modèle. Les quatre variables du modèle sont la fréquence f , la quantification q , le délai global d et la vitesse de la tête s . La métrique QoE correspond à la valeur de la fonction spline $S(f, q, d, s)$.

Nous faisons l'hypothèse que la QoE diminue de manière monotone lorsque le délai augmente, la fréquence diminue ou la quantification diminue. Nous représentons ce comportement en fixant la monotonie de la spline, c'est à dire en contraignant la croissance/décroissance des coefficients correspondants (Sect. A.6).

La spline est définie dans un domaine borné par les valeurs minimales et maximales de chacune des quatre variables. La fréquence f est comprise arbitrairement entre 1 et 100 Hz correspondant à une large gamme de fluidité atteignable sur des équipements récents. La quantification q correspond au paramètre de compression d'un encodeur fictif qui s'étend de 0 à 100, où la valeur de 100 engendre une image identique à l'image de référence alors que la valeur de 0 donne une image avec moins de détails et donc de moins bonne qualité. Comme il n'est pas fait référence à un encodeur particulier, la notion de qualité de l'image encodée est très relative. Le délai d est pris en compte entre 0 et 100 ms et la vitesse de rotation de la tête s entre 0 et $1 \frac{\text{rad}}{\text{s}}$ (≈ 60 degrés par seconde), comme suggéré dans de précédentes études [19].

Le modèle générique proposé correspond à une spline d'ordre 3 pour toutes les variables ; il s'agit d'un compromis entre la flexibilité et le nombre de coefficients nécessaires. Le modèle générique peut être coupé dans l'espace d'encodage augmenté en sections, pour obtenir le modèle pour un contexte donné : il s'agit du modèle contextuel $M_c(f, q, k)$ (Fig. 4). Des noeuds sont placés aux extrémités de chaque variable avec une multiplicité égale à l'ordre. Les noeuds intérieurs sont placés arbitrairement ; une densité élevée augmente la souplesse.

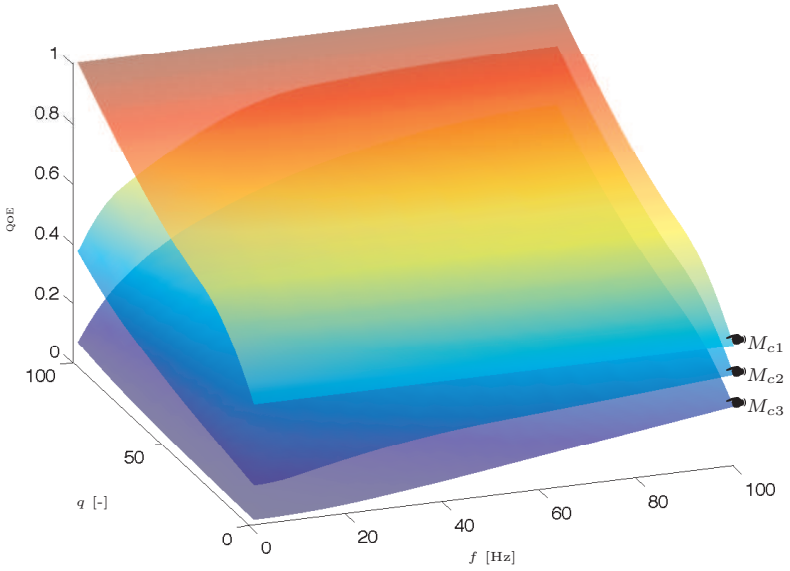
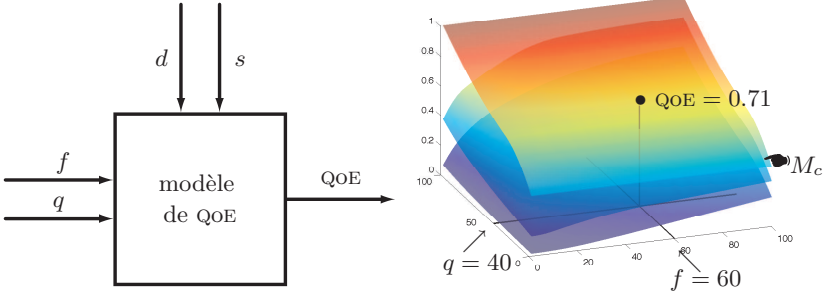


Fig. 4: Le modèle contextuel $M_c(f, q, k)$ correspond à la section de coupe du modèle générique dans l'espace d'encodage augmenté pour un contexte donné. Trois modèles M_{ci} sont représentés pour $(\bar{d}_1 = 0, \bar{s}_1 = 0)$, $(\bar{d}_2 = 10, \bar{s}_2 = 0.2)$, $(\bar{d}_3 = 20, \bar{s}_3 = 0.4)$.

Le modèle générique M_g est donc donné par la fonction spline multivariable suivante :

$$M_g(f, q, d, s) = S(f, q, d, s) \quad (3)$$

avec les noeuds et les coefficients choisis arbitrairement (Annexe B). La Sect. 2.4.2 présente l'identification des coefficients α du modèle M à partir de mesures expérimentales.



(a) Le modèle QoE permet de déterminer la valeur de la métrique QoE à partir des valeurs des paramètres d'encodage f , q et des perturbations contextuelles d et s .

(b) Le modèle contextuel M_c pour le contexte $d = 0$ et $s = 0$ permet de trouver la valeur de la QoE, pour des paramètres d'encodage f et q donnés.

Fig. 5: La valeur de la métrique QoE est obtenue à partir du modèle et des valeurs des variables.

2.3.2 Simulation du modèle générique

Cette section montre comment la métrique QoE est évaluée en temps réel, lorsque les variables du modèle de QoE f , q , d et s varient. Le modèle générique M_g (Sect. 2.3 et Annexe B) est utilisé pour déterminer la valeur de la QoE. A chaque instant k , le modèle contextuel M_c est sélectionné en fonction des valeurs des variables contextuelles s et d . Les valeurs des paramètres d'encodage f et q donnent la valeur de la QoE, visible sur le modèle contextuel (Fig. 5).

L'estimation de la QoE est illustrée avec une simulation lors de laquelle les variables évoluent au cours du temps. Les Fig. 6 et 7 montrent comment la métrique QoE est calculée. La Fig. 6 présente l'évolution temporelle de variables f , q , d et s ainsi que la métrique

QoE. La Fig. 7 montre les modèles contextuels sélectionnés pour des instants donnés.

La simulation se déroule en deux phases. Dans la première phase jusqu'à l'instant ②, les variables contextuelles sont constantes et les paramètres d'encodage varient. Dans la deuxième phase à partir de l'instant ②, les paramètres d'encodage sont constants et les variables contextuelles varient.

Paramètres d'encodage variables

La simulation commence avec une vitesse de tête lente $s = 0.1 \frac{\text{rad}}{\text{s}}$ et un délai faible $d = 10 \text{ ms}$. Le modèle contextuel M_c correspondant est constant (Fig. 7a) jusqu'à l'instant ②. Il provient de la coupe du modèle générique M_g dans l'espace d'encodage augmenté.

À l'instant ①, la fréquence $f=1 \text{ Hz}$ et le facteur de quantification $q = 0$ sont à leurs valeurs minimales (Fig. 6c). La valeur de la métrique QoE est calculée à partir du modèle contextuel (Fig. 7a), puis reportée dans le graphique correspondant (Fig. 6d).

Durant la période ①, la fréquence augmente progressivement jusqu'à sa valeur maximale $f = 100 \text{ Hz}$. La valeur de la métrique QoE augmente (Fig. 6d). Cela correspond à suivre l'arête du modèle contextuel M_c le long de l'axe de la fréquence f (Fig. 7a).

De manière similaire durant la période ②, le facteur de quantification augmente jusqu'à $q = 100$. La valeur de la métrique QoE augmente encore. Cela correspond à suivre l'arête du modèle contextuel M_c le long de l'axe de la quantification q .

Durant la période ③, la fréquence et la quantification sont diminuées jusqu'à $f = 50 \text{ Hz}$ et $q = 50$. La valeur de la métrique QoE diminue. Cela correspond à se déplacer vers le centre du modèle contextuel M_c .

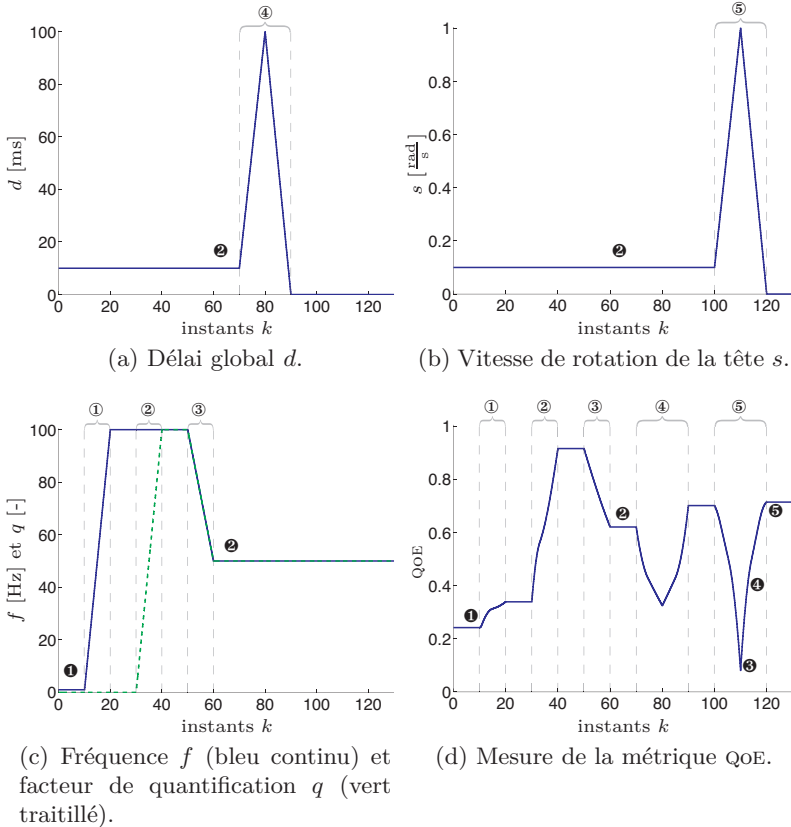
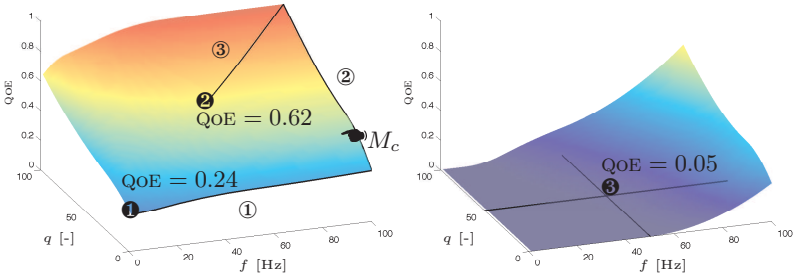
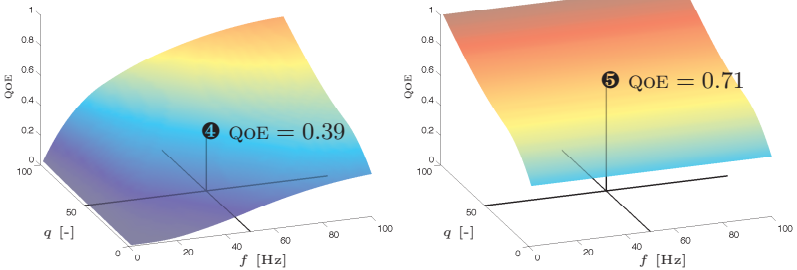


Fig. 6: Les variables contextuelles (le délai d et la vitesse de la tête s) et les paramètres d'encodage (la fréquence f et la quantification q) évoluent arbitrairement. La valeur de la métrique QoE est obtenue grâce au modèle générique M_g . Les instants sont notés avec un nombre blanc sur un disque noir. Les périodes sont notées avec un nombre noir sur un disque blanc.



(a) Entre les instants ❶ et ❷, le modèle contextuel M_c est sélectionné pour les valeurs des variables contextuelles $s = 0.1$ et $d = 10$.

(b) A l'instant ❸, le modèle contextuel M_c est sélectionné pour les valeurs des variables contextuelles $s = 1$ et $d = 0$.



(c) A l'instant ❹, le modèle contextuel M_c est sélectionné pour les valeurs des variables contextuelles $s = 0.5$ et $d = 0$.

(d) A l'instant ❺, le modèle contextuel M_c est sélectionné pour les valeurs des variables contextuelles $s = 0$ et $d = 10$.

Fig. 7: Le modèle contextuel M_c , correspondant à la coupe du modèle générique M_g dans l'espace d'encodage augmenté, est représenté pour différents instants.

Contexte variable

A partir de l'instant ②, les deux paramètres d'encodage restent constants. Par contre, les variables contextuelles d et s varient. L'allure du modèle contextuel M_c évolue au cours du temps.

Durant la période ④, la valeur du délai change. La QoE diminue quand le délai augmente et vice versa.

Durant la période ⑤, la vitesse de la tête change. A l'instant ③, la vitesse de la tête est élevée $s = 1 \frac{\text{rad}}{\text{s}}$. Le modèle contextuel correspondant est illustré dans la Fig. 7b. Pour des paramètres d'encodage identiques à précédemment, la valeur de la métrique QoE est nettement plus faible. Ceci est dû au décalage entre les mondes réel et virtuel engendré par la relativement basse fréquence d'affichage. A l'instant ④, la vitesse de la tête diminue à $s = 0.5 \frac{\text{rad}}{\text{s}}$ et la valeur de la QoE augmente. A l'instant ⑤, la tête est statique et la QoE résultante atteint une valeur encore plus élevée.

2.4 Identification du modèle

Le modèle générique de QoE présenté dans la Sect. 2.3.1 ne considère pas les conditions d'utilisation. Le facteur de quantification n'est par exemple pas lié à un encodeur. Le modèle de QoE est donc identifié pour les conditions d'utilisation particulières données, sur la base d'évaluations subjectives de l'expérience utilisateur. L'idée est de récolter une mesure quantitative représentant la satisfaction de l'utilisateur pour différentes combinaisons de variables, afin de saisir l'impact de la quantification, de la fréquence des images, du délai et de la vitesse angulaire de la tête sur l'expérience utilisateur.

Cette section présente la procédure pour identifier le modèle de QoE. La Sect. 2.4.1 présente la méthode d'évaluation subjective qui permet d'obtenir les appréciations subjectives de l'utilisateur pour les différentes combinaisons de variables. La Sect. 2.4.2 montre comment trouver l'hypersurface qui approxime au mieux les points de mesure. L'hypersurface est la spline multivariable qui représente le

modèle identifié. La Sect. 2.4.3 présente l'identification expérimentale du modèle par un utilisateur pilote, afin de confirmer l'allure de l'hypersurface choisie.

2.4.1 Evaluation subjective

La première étape de l'identification de modèle concerne l'évaluation subjective. Il est demandé à l'utilisateur de donner son appréciation pour différentes valeurs de chaque variable.

Un ensemble de valeurs est choisi pour chacune de s variables. Pour la fréquence f , l'ensemble des valeurs x_f est donné par le *vecteur d'entrée* $[f_0, \dots, f_{n_f}]$. Les ensembles pour la quantification x_q , le délai x_d et la vitesse de la tête x_s sont construits de manière similaire. Les valeurs de ces quatre ensembles peuvent être combinées sous la forme d'une *matrice d'entrées* x comprenant quatre colonnes (nombre de variables prises en comptes) et $n_M = \prod_{i=f,q,d,s} n_i$ lignes (nombre de combinaisons différentes possibles).

Des tests subjectifs sont ensuite effectués sur un groupe d'utilisateurs test, en s'inspirant des recommandations de l'ITU [42, 43]. L'utilisateur est équipé d'un système d'AR puis est confronté aux différentes combinaisons de variables contenues dans x , présentées dans un ordre aléatoire. Les essais durent environ 10 secondes et sont séparés par une pause lors de laquelle l'utilisateur doit donner une note comprise dans l'intervalle $[0, 1]$ (ou une autre échelle plus parlante pour l'utilisateur) pour représenter son appréciation. La note 1 correspond à la meilleure expérience. Il est important de noter que l'échelle utilisée n'offre pas une représentation absolue, mais seulement une information relative entre les valeurs. Ainsi, une note de 0.6 correspond à une expérience meilleure qu'une note de 0.3, mais ça ne veut pas dire que l'expérience est deux fois plus élevée. Les notes sont récoltées dans un *vecteur colonne de sortie* y où leur position correspond à la combinaison représentée dans la matrice d'entrées x . Les n points de mesures correspondent aux tuples $(x_{(i,\bullet)}, y_i)$.

Certaines variables peuvent être fixées manuellement, comme c'est le cas pour le délai, la fréquence et la quantification dont les valeurs sont modifiables directement dans un environnement de test. D'autres variables ne peuvent pas être imposées techniquement. C'est le cas de la vitesse de rotation de la tête qui dépend des mouvements de l'utilisateur. Il est cependant possible de contraindre l'utilisateur à tourner la tête à une vitesse désirée en lui demandant d'orienter sa tête entre deux points fixes en une période de temps fixe. On obtient ainsi un déplacement angulaire de la tête quasi-triangulaire [25]. Un métronome digital audio est utilisé pour donner la période de temps à l'utilisateur.

La durée du test subjectif pour un utilisateur dépend du nombre d'appréciations demandées. Il faut évidemment y ajouter le temps nécessaire à installer l'utilisateur et lui expliquer le déroulement de l'expérimentation. Dans une période d'une heure, il est possible d'obtenir environ 150 appréciations. Il est recommandé de ne pas tenir compte des 10 premières appréciations lors desquelles l'utilisateur s'accoutume à l'environnement [42, 43]. De plus, afin de diminuer l'effet des erreurs d'appréciation, chaque combinaison doit être notée deux fois par l'utilisateur. En tenant compte de ces remarques, on aperçoit qu'environ $n_M = 70$ combinaisons différentes peuvent être montrées à l'utilisateur en une heure.

2.4.2 Hypersurface d'approximation

Le but de l'approximation est de trouver une hypersurface qui passe au plus près des points de mesure x, y . Comme la fonction spline multivariable S a été choisie pour représenter cette hypersurface, le modèle $M(f, q, d, s)$ est entièrement défini par les noeuds et les coefficients de la spline. L'ordre de la spline est choisi pour un compromis de flexibilité. Un ordre élevé permet d'avoir une plus grande flexibilité entre deux noeuds donnés, mais requiert plus de points pour être défini entre ces deux noeuds. Un ordre de 3 permet d'obtenir une flexibilité suffisante même pour un faible nombre de points de mesure.

Les noeuds τ de la spline S sont choisis en fonction de l'ordre et des vecteurs d'entrée x_f , x_q , x_d et x_s . Pour chacune des dimensions, des noeuds sont placés aux extrémités du vecteur d'entrée avec une multiplicité égale à l'ordre (3) afin que la valeur et la dérivée de la spline puissent y être non nulles. Des noeuds intérieurs sont positionnés de telle sorte qu'il y ait au moins un noeud entre chaque paire de valeurs successives. Par exemple si la dimension de la fréquence contient un vecteur d'entrée à quatre valeurs $x_f = [f_0, f_1, f_2, f_3]$, les noeuds pour la fréquence seront $\tau_f = [\tau_{f,0}, \tau_{f,1}, \tau_{f,2}, \tau_{f,3}, \tau_{f,4}, \tau_{f,5}, \tau_{f,6}]$, tels que $\tau_{f,0} = \tau_{f,1} = \tau_{f,2} = f_0$, $\tau_{f,4} = \tau_{f,5} = \tau_{f,6} = f_3$ et $f_1 < \tau_{f,3} < f_2$.

Les coefficients α de la spline S sont déterminés grâce à une procédure d'optimisation qui minimise l'erreur au carré entre la spline et les points de mesure. Cependant, une contrainte est introduite pour forcer la monotonie dans la directions des variables : la métrique QoE augmente quand la fréquence augmente, quand la quantification augmente et quand le délai diminue.

$$\underset{\alpha}{\operatorname{argmin}} \sum_{i=1}^{n_M} (y_i - S(x_{(i,\bullet)}))^2 \quad (4)$$

$$\alpha_{(\zeta_f, \dots)} \leq \alpha_{(\zeta_f+1, \dots)} \quad (5)$$

$$\alpha_{(\dots, \zeta_q, \dots)} \leq \alpha_{(\dots, \zeta_q+1, \dots)} \quad (6)$$

$$\alpha_{(\dots, \zeta_d, \dots)} \geq \alpha_{(\dots, \zeta_d+1, \dots)} \quad (7)$$

Un algorithme d'optimisation standard (par exemple la fonction `fmincon` de MATLAB) permet de trouver les valeurs des coefficients α .

2.4.3 Identification expérimentale

Afin de valider la procédure d'identification et de confirmer l'allure du modèle, notamment le comportement monotone sur certaines variables, nous avons appliqué la procédure d'identification pour un utilisateur pilote. Ce dernier n'avait jamais été confronté à une application d'AR auparavant.

Le prototype d'AR employé pour cette identification expérimentale possède des limitations qui ne permettent pas d'identifier un modèle pour une large nombre d'utilisateurs. Le délai minimum est de l'ordre de 80 ms et la fréquence maximale d'environ 30 Hz. Ainsi, la plage de mesure disponible pour ces variables ne permet pas d'obtenir un modèle suffisamment complet.

Matériel

Le modèle de QoE a été identifié dans le cadre d'un scénario particulier avec un matériel donné. Dans le scénario considéré, la scène AR consiste en un pendule inversé virtuel immobile posé sur une table. L'utilisateur peut bouger sa tête librement et regarder le pendule affiché dans son HMD à transparence optique. Le HMD monoculaire (Liteye 750, <http://www.liteye.com>) affiche les images à une résolution de 800×600 pixels. Son champ de vision de 22 degrés permet à l'utilisateur d'avoir un bon sentiment d'immersion. Un système de positionnement hybride inertio-optique (InterSense IS-1200 VisTracker, <http://www.intersense.com>) est fixé sur le HMD. Une constellation de marqueurs est placée au plafond de la zone de test. La *pose* est calculée sur la base des images acquises par la caméra et des mesures de la plate-forme inertielle. Basé sur la mesure du point de vue de l'utilisateur, la scène virtuelle est générée sur un ordinateur. L'image est ensuite affichée sur le HMD connecté à l'ordinateur. L'ensemble placé sur la tête de l'utilisateur, comprenant le système de positionnement et le HMD, a une masse d'environ 500 g.

Plage de mesure

Dans ce scénario, trois variables ont été prises en compte. La fréquence de génération de l'image f , la quantification de l'image q et la vitesse de rotation de la tête s . L'impact du délai d n'est pas mesuré durant cette identification expérimentale ; il est supposé constant à environ 80 ms (borne inférieure imposé par le prototype

utilisé). Trois fréquences ont été choisies dans l'intervalle possible : 2 Hz, 10 Hz et 20 Hz. Pour ce scénario, la quantification de l'image se fait par la résolution de l'image générée. Trois résolutions différentes ont été sélectionnées : 800×600 , 438×329 (30% de la résolution 800×600), 252×190 (10% de la résolution 800×600). L'image résultante est ensuite mise à l'échelle pour la résolution d'affichage de 800×600 . Trois vitesses de rotation de la tête ont été sélectionnées : quasi-statique, $0.2 \frac{\text{rad}}{\text{s}}$ et $0.4 \frac{\text{rad}}{\text{s}}$.

Nous avons donc les vecteurs de variables suivants :

$$\begin{aligned}x_f &= [2, 10, 20] \\x_q &= [0.1, 0.3, 1] \\x_s &= [0, 0.2, 0.4]\end{aligned}$$

et obtenons une matrice d'entrée x de 27 lignes et 3 colonnes. L'utilisateur est confronté deux fois à chacune des combinaisons de cette matrice, dans un ordre aléatoire.

Environnement

L'utilisateur est assis sur une chaise fixe à 140 cm de la table où l'objet virtuel est positionné (Fig. 8). Deux marqueurs sont physiquement placés sur la table avec un écart de 70 cm. Les marqueurs correspondent à l'emplacement des extrémités du rail du pendule virtuel et donnent ainsi la position absolue de l'objet virtuel. L'utilisateur observe la table réelle augmentée du pendule virtuel. Le mouvement de la tête est imposé en demandant à l'utilisateur de tourner la tête entre deux points fixe de la table, en gardant le regard fixe par rapport à la tête. Le mouvement correspond à un angle de lacet de 40 degrés, effectué en une période de temps donnée par le métronome audio [25]

Résultats

Après chaque essai, l'utilisateur donne une note d'appréciation de l'expérience utilisateur comprise entre 0 et 1, où 1 correspond à

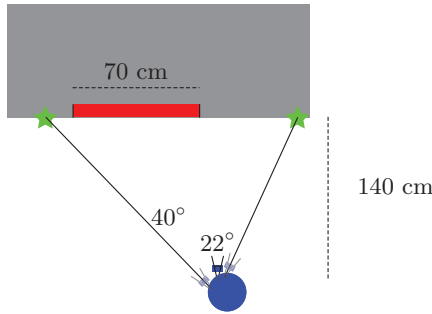


Fig. 8: L'utilisateur est positionné à un endroit fixe et tourne la tête pour observer un objet virtuel placé sur une table. Le pendule (bande allongée rouge) est positionné sur la table (rectangle gris). L'angle de vue du HMD est de 22 degrés.

la meilleure expérience utilisateur. Ces valeurs sont stockées dans le vecteur de sortie y .

Le Tableau 1 montre la matrice d'entrée x et le vecteur sortie y pour les 54 mesures prises en compte pour l'identification, dans l'ordre appliqué.

La spline multivariable d'ordre 3 représente le modèle. Les noeuds sont placés aux extrémités :

$$\begin{aligned}\tau &= \{\tau_f, \tau_q, \tau_s\} \\ \tau_f &= [2, 2, 2, 20, 20, 20] \\ \tau_q &= [0.1, 0.1, 0.1, 1, 1, 1] \\ \tau_s &= [0, 0, 0, 0.4, 0.4, 0.4]\end{aligned}$$

Les données exhibent un comportement monotone dans la direction de la fréquence f et celle de la quantification q . L'allure monotone de la fonction spline est donc forcée dans ces deux directions lors de l'optimisation. Le tableau de coefficients α à trois dimensions

x_1	y_1	x_2	y_2
2 0.3 0.2	0.3	2 0.1 0	0.3
10 0.3 0	0.4	10 0.3 0.2	0.6
2 1 0	0.7	10 0.3 0.4	0.5
2 1 0.2	0.5	20 0.3 0.2	0.6
20 1 0.4	0.9	2 0.3 0.4	0.3
2 0.1 0.4	0.1	20 0.1 0.2	0.5
10 1 0.4	0.8	20 0.3 0.4	0.6
20 0.3 0.4	0.6	10 1 0.4	0.6
2 0.3 0.4	0.2	2 1 0.2	0.4
10 0.1 0.2	0.5	10 1 0	1
10 0.3 0.4	0.6	2 0.3 0	0.5
2 0.1 0	0.4	20 0.3 0	0.7
10 0.3 0.2	0.6	20 1 0.4	0.8
20 0.1 0.2	0.6	20 1 0.2	0.9
20 0.1 0	0.4	2 0.1 0.2	0.1
20 0.1 0.4	0.5	2 1 0.4	0.4
20 1 0.2	0.8	10 0.3 0	0.5
2 0.1 0.2	0.1	2 0.3 0.2	0.4
20 1 0	0.9	20 1 0	1
10 0.1 0	0.4	10 0.1 0	0.4
2 0.3 0	0.6	2 1 0	0.8
20 0.3 0	0.7	20 0.1 0	0.4
20 0.3 0.2	0.7	10 1 0.2	0.8
10 1 0	0.8	10 0.1 0.2	0.5
10 1 0.2	0.7	20 0.1 0.4	0.5
2 1 0.4	0.3	10 0.1 0.4	0.4
10 0.1 0.4	0.5	2 0.1 0.4	0.2

Tableau 1: Evaluation subjective. Les combinaisons de paramètres sont placés dans la matrice d'entrée $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ avec les colonnes représentant les paramètres f , q et s . Les appréciations de l'utilisateur forment le vecteur de sortie correspondant $y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$.

issu de l'approximation pour les noeuds τ et les mesures x et y est présenté dans le Tableau 2.

Le modèle identifié $M_i = S(f, q, s)$ est défini par les noeuds τ et les coefficients α . Son allure est présentée dans la Fig. 9 dans l'espace d'encodage augmenté pour cinq vitesses de tête.

$$\alpha_{(\bullet,\bullet,1)} = \begin{pmatrix} 0.3782 & 0.6682 & 0.7488 \\ 0.3808 & 0.6682 & 0.9602 \\ 0.4470 & 0.9713 & 0.9713 \end{pmatrix}$$

$$\alpha_{(\bullet,\bullet,2)} = \begin{pmatrix} 0.0875 & 0.4174 & 0.4174 \\ 0.6552 & 0.6652 & 0.8227 \\ 0.6575 & 0.7459 & 0.8227 \end{pmatrix}$$

$$\alpha_{(\bullet,\bullet,3)} = \begin{pmatrix} 0.1640 & 0.3624 & 0.3624 \\ 0.5424 & 0.6889 & 0.8486 \\ 0.5424 & 0.7064 & 0.8486 \end{pmatrix}$$

Tableau 2: Tableau des coefficients α issu de l'optimisation sous contraintes. Les dimensions représentent dans l'ordre la fréquence f , la quantification q et la vitesse de la tête s .

Discussion

L'identification expérimentale a permis de valider la procédure d'évaluation subjective. Lors de cette expérience, nous avons remarqué que l'utilisateur a donné des notes très proches pour toutes les combinaisons de paramètres montrées deux fois. Cela suggère que pour un utilisateur donné, l'évaluation subjective est reproductible.

L'expérimentation a duré environ 35 minutes, conformément à ce qui était attendu. En cas de durée plus longue, il semble nécessaire de faire une pause pour garder la concentration de l'utilisateur. Avant l'expérimentation, il est important de sensibiliser l'utilisateur à l'environnement de test, et de lui faire prendre conscience intuitivement qu'il doit évaluer la fluidité et le niveau de détail.

L'application et le matériel utilisés pour le test comportent des limitations qui n'ont pas permis d'obtenir des mesures plus nombreuses. Le système de positionnement engendre une erreur de positionnement statique. Avec le logiciel utilisé, il n'est pas possible d'obtenir un délai global faible. Il n'aurait donc pas été possible

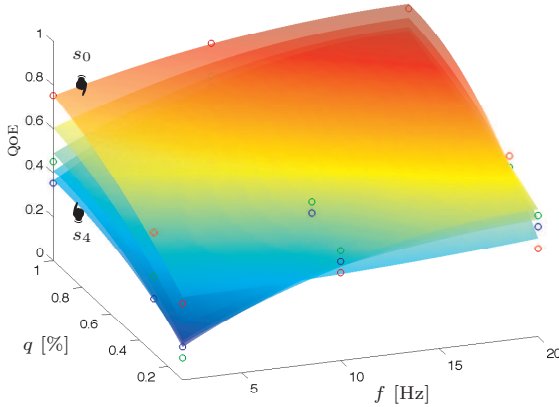


Fig. 9: Le modèle de QoE $M_i(f, q, s)$ est identifié pour trois variables. Les mesures sont représentés par des cercles dans l'espace d'encodage augmenté. Le modèle est représenté pour cinq contextes différents : vitesses de tête s_0 quasi-statique, à $s_1 = 0.1$, $s_2 = 0.2$, $s_3 = 0.3$ et $s_4 = 0.4 \frac{\text{rad}}{\text{s}}$.

d'évaluer le modèle pour des délais inférieurs ; il serait évidemment intéressant de pouvoir évaluer l'impact du délai étant donné qu'il influence fortement la QoE. De même, le temps de calcul nécessaire à la création des images ne permet pas d'atteindre une fréquence d'images élevée. Malgré ces limitations, il a tout de même été possible d'identifier le modèle pour une plage réduite des variables.

2.5 Conclusion

Dans ce chapitre, nous avons modélisé l'expérience utilisateur dans le contexte de l'AR mobile. Nous avons d'abord défini la métrique QoE pour l'évaluation objective de l'expérience utilisateur. La

valeur de cette métrique est estimée en temps réel à partir d'un modèle de QoE et des mesures des variables choisies ayant un impact sur l'expérience utilisateur. Ces variables ont été identifiées pour l'AR mobile. Nous avons pris en compte deux paramètres d'encodage (la fréquence et la quantification) ainsi que deux variables contextuelles (la vitesse de la tête et le délai global). Le modèle est exprimé comme une fonction spline multivariable de ces variables. Ce modèle est coupé en sections dans l'espace d'encodage augmenté pour obtenir le modèle contextuel qui ne dépend plus que des paramètres d'encodage. L'utilisation du modèle générique est illustré au travers d'une simulation où la métrique QoE est estimée pour différents profils de paramètres d'encodage et de variables contextuelles. Finalement, le modèle de QoE est identifié pour un cas particulier grâce à une évaluation subjective requérant l'utilisateur. Ce modèle identifié confirme les hypothèses concernant l'allure de la surface. Il peut dès lors être utilisé pour l'adaptation sans avoir recours à l'utilisateur.

La métrique objective perceptuelle proposée et le modèle de QoE sont inspirés des études sur l'évaluation de la qualité de flux vidéo en y intégrant les notions propres à l'AR (voir état de l'art aux Sect. 2.1.1 et 2.1.2). La structure mathématique proposée permet de considérer plusieurs variables, ce que les modèles existants ne permettent généralement pas. Comme suggéré par Cranley et al. [16], les modèles de qualité peuvent être utilisés pour l'adaptation des paramètres d'encodage ; c'est le cas du modèle de QoE proposé.

Le modèle de QoE est présenté pour l'AR mobile, mais la méthodologie peut s'appliquer à d'autres domaines d'utilisation. Dans ce cas, il faut identifier les variables qui modifient l'expérience utilisateur et séparer les variables modifiables (dans notre cas les paramètres d'encodage) des variables qui ne peuvent pas être modifiées mais qui sont mesurables (comme les variables contextuelles).

Le modèle développé dans ce chapitre sera utilisé dans le schéma d'adaptation présenté dans le Chap. 5. Ce modèle permettra de déterminer les valeurs à appliquer en temps réel pour les paramètres d'encodage, en fonction de l'estimation des variables contextuelles.

Modèle pour le contrôle de débit

Le système encodeur est représenté par un modèle abstrait pour permettre la conception d'un contrôle de débit linéaire.

Les données d'AR sont transmises vers l'utilisateur sous forme de flux vidéo au travers d'un canal de communication. Les capacités du canal de communication limitent le débit¹ de transmission. En fonction de l'infrastructure ou du protocole utilisés, le flux vidéo peut subir des délais et/ou des pertes de données. Il devient nécessaire de contrôler le débit du flux vidéo par rapport au débit disponible du canal de communication pour assurer la meilleure QoE possible.

L'encodage du flux vidéo permet de modifier son débit. Cependant, le *contenu* du flux vidéo peut également influencer son débit. Par exemple pour une compression spatiale, une image avec de nombreux détails aura une taille plus grande qu'une image de couleur uniforme. De même si l'encodeur utilise une compression temporelle, une séquence avec de rapides changements engendrera un débit plus élevé qu'une séquence vidéo statique.

Un mécanisme de contrôle de débit² doit être mis en place pour gérer le débit du flux vidéo. Le contrôle de débit adapte les paramètres d'encodage du flux vidéo en fonction des capacités du canal de communication et du contenu.

1. *rate*

2. *rate control*

Dans ce chapitre, nous proposons un modèle linéaire monovariable de l'encodeur qui permet de concevoir un contrôleur. Le modèle considère plusieurs paramètres d'encodage simultanément de façon abstraite. Afin d'obtenir la meilleure expérience utilisateur possible, les paramètres d'encodage seront sélectionnés ultérieurement en utilisant le modèle de QoE développé dans le Chap. 2.

Une approche système est utilisée pour modéliser le procédé d'encodage de manière générique. L'effet des paramètres d'encodage est représenté par une nouvelle entrée unique u . Le modèle abstrait ainsi obtenu présente un comportement linéaire, simplifiant ainsi le développement du contrôle de débit. Le Chap. 4 présente la méthode qui permet ensuite d'appliquer le contrôle de débit à un encodeur avec ses propres paramètres d'encodage.

La Sect. 3.1 rappelle brièvement les notions de transmission de données et présente d'autres approches pour le contrôle de débit. L'approche système est ensuite présentée dans la Sect. 3.2. Les systèmes encodeur et transmission y sont introduits. La Sect. 3.3 présente le nouveau modèle générique de l'encodeur où le contenu apparaît sous forme de perturbation. Finalement, la Sect. 3.4 rappelle les éléments principaux et resitue le présent chapitre dans la thèse.

3.1 Etat de l'art

Habituellement, le contrôle de débit est un mécanisme qui traite à la fois l'encodage et l'envoi du flux vidéo. Il adapte les paramètres d'encodage pour contrôler le niveau du tampon³ de transmission. Les données contenues dans ce tampon sont envoyées quand le canal de communication le permet, en fonction du protocole de communication utilisé. Le tampon rend ainsi la transmission plus fluide, ce qui permet de limiter la gigue⁴ du flux vidéo au minimum et d'utiliser un mécanisme de correction d'erreur. Cependant, le tampon

3. *buffer*

4. *jitter*

engendre généralement un délai non négligeable entre la création du flux vidéo et son envoi. Dans le cas du *streaming* (diffusion de flux continu) vidéo par exemple, le tampon a un effet bénéfique car il est important pour l'utilisateur de recevoir un flux vidéo fluide alors que le délai pouvant atteindre plusieurs secondes n'est pas gênant. Le mécanisme de contrôle de débit s'occupe de réguler le remplissage du tampon. Lorsque le niveau du tampon diminue ou atteint un seuil bas, le contrôle de débit augmente le débit du flux vidéo pour alimenter le tampon. Lorsque le niveau du tampon augmente ou atteint un seuil élevé, le contrôle de débit diminue le débit du flux vidéo.

Dans les prochains paragraphes, nous présentons d'abord des stratégies de contrôle de débit qui tiennent compte de la qualité perçue du flux vidéo présenté à l'utilisateur. D'autres études mettent en oeuvre des contrôleurs au sens automatique pour améliorer les performances du contrôle de débit. Finalement, nous proposons une approche générique du contrôle de débit qui permet d'être indépendant du type d'encodeur et du mécanisme de transmission.

3.1.1 Contrôle de débit avec notion de qualité

Saw présente les principes de compression vidéo intégrant la notion de qualité ainsi que les techniques conventionnelles de contrôle de débit [44]. L'auteur montre d'abord la nature non stationnaire de la vidéo ; la taille de la vidéo dépend de son contenu. La base de la compression vidéo consiste à supprimer les redondances spatiales et temporelles et le fractionnement des macroblocs de l'image comme c'est le cas pour la compression MPEG (Moving Picture Experts Group). Les techniques de contrôle du débit-qualité compressent la vidéo en tenant compte de métriques objectives afin de garantir une certaine qualité de la vidéo. L'auteur présente également des approches plus originales comme celle basée sur la prédiction linéaire et la quantification non linéaire, ou les approches de logique floue.

Chen et Ngan présentent les récentes évolutions du contrôle de débit pour le codage vidéo [45]. Le contrôle de débit tend à obtenir

une qualité d'image maximale sans excéder le débit disponible. La qualité est représentée avec des mesures de distorsion comme le rapport crête signal sur bruit (PSNR pour *Peak Signal to Noise Ratio*) qui ne sont pas liées au système visuel humain (HVS pour *Human Visual System*) [12]. Toujours selon Chen et Ngan, le principe de contrôle de débit est d'ajuster les paramètres de quantification de chaque image pour réduire le débit de l'image compressée. Pour une compression en temps réel, les algorithmes gourmands en puissance de calculs doivent être évités pour respecter les contraintes de délai et de gigue. Les algorithmes à une passe⁵ sont ainsi préférés aux algorithmes à plusieurs passes⁶. Les fondements théoriques du contrôle de débit basés sur la théorie débit-distorsion montrent que les algorithmes recommandés par les standards de codage vidéo ne correspondent pas à la solution optimale. De plus, les algorithmes recommandés ne font pas partie de la norme des standards vidéo étant donné que de nombreux schémas de contrôle de débit sont spécifiques à un standard, comme H.261, MPEG-1 ou MPEG-4. Les auteurs soulignent le fait que le domaine du contrôle de débit laisse encore place à l'amélioration. Ils donnent quelques pistes concernant des techniques subjectives correspondant mieux à la perception humaine.

3.1.2 Contrôle de débit basé sur l'automatique

Il existe également des conceptions de contrôleurs pour le contrôle de débit de flux en *streaming* basés sur les théories de l'automatique. Yu et Ahmad ont été les premiers à utiliser un contrôleur PID (Proportionnel Intégral Dérivé) pour obtenir un débit précis d'un flux MPEG-4 tout en gardant la mémoire tampon à un niveau adéquat [46]. Les auteurs ont remarqué que les solutions de contrôle de débit comme le VM8 (Verification Model, version 8) n'utilisaient jusqu'alors qu'un contrôleur proportionnel pour la mémoire tampon.

5. *single-pass*

6. *multi-pass*

Ils ont donc logiquement étendu le contrôleur à la forme PID pour obtenir un contrôle de débit évitant les débordements du tampon et offrant une qualité vidéo prometteuse. D'autres recherches ont été menées pour l'utilisation de contrôleurs pour gérer la compression vidéo, que ce soit pour gérer le compromis de la qualité spatio-temporelle du TMN8 (*video codec* Test Model, Near-term, version 8) avec un PID lors de contraintes de délai [47] ou pour éviter la variation de la qualité lors de changement de scène enregistrée pour le standard H.264 avec un algorithme PID [48].

Ces approches restent cependant spécifiques à une norme ou un encodeur et ne peuvent pas être utilisées d'une façon générique. Bruna et al. propose un nouvel algorithme pour les images JPEG (Joint Photographic Experts Group) qui consiste à calculer le nouveau facteur de qualité à partir du facteur de qualité précédent et du rapport entre le débit de référence et le débit mesuré [49]. L'algorithme offre une convergence rapide et peut être utilisé pour l'encodage de flux vidéos M-JPEG (Motion Joint Photographic Experts Group). Cependant, les auteurs ne proposent pas de stratégie pour gérer plusieurs paramètres d'encodage de manière générique, ni d'analyse de stabilité.

3.1.3 Approche générique du contrôle de débit

Les stratégies de contrôle de débit conventionnelles sont généralement spécifiques à un type d'encodeur et protocole de communication. Comme il n'y a pas de consensus pour la transmission de flux vidéo d'AR, nous proposons une approche plus générique qui n'est pas spécifique au choix d'implémentation.

Notre approche du contrôle de débit considère le flux vidéo jusqu'à qu'il soit encodé. Le mécanisme de transmission n'est pas spécifié, mais le flux vidéo doit être envoyé directement après son encodage, en laissant le tampon vide. Le délai dû au tampon est ainsi supprimé. Nous proposons notre propre définition pour le contrôle de débit.

Définition 5 (Contrôle de débit). *Le contrôle de débit est un mécanisme situé au niveau de l'encodage. Ce mécanisme manipule les paramètres d'encodage en temps réel pour que le débit du flux encodé suive un débit de référence.*

3.2 Approche système

L'utilisation d'une approche système pour représenter le contrôle de débit permet d'obtenir une vision abstraite rendant l'utilisation des outils de l'automatique possible. Le concept de système correspond à un ensemble d'entités qui interagissent. Un système dynamique est un processus avec des entrées qui modifient son comportement et des sorties qui illustrent l'impact de ces modifications (Fig. 10). Les entrées et les sorties d'un système sont généralement choisies en lien avec la physique ou l'implémentation (par exemple en mécanique, la tension appliquée aux bornes d'un moteur est une entrée naturelle), mais elles peuvent aussi représenter des notions plus abstraites. Le système évolue dans le temps en suivant certaines lois qui peuvent être regroupées dans un modèle du système.

Les systèmes peuvent être connectés entre eux si l'interopérabilité le permet. Plusieurs systèmes interconnectés forment à nouveau un système. De même, le traitement d'un système peut être facilité en le découpant en sous-systèmes interconnectés. Le niveau d'abstraction utilisé pour représenter un processus est déterminé en



Fig. 10: Un système est représenté sous forme de bloc. Les entrées modifient le comportement d'un système dynamique. Les sorties reflètent ces modifications.

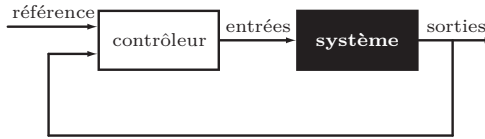


Fig. 11: Le contrôleur (bloc blanc) modifie les entrées en fonction des valeurs de sortie et des valeurs de références pour contrôler le système (bloc noir). C'est le principe de la boucle de rétroaction.

fonction de l'objectif que l'on veut atteindre. Une abstraction trop élevée ne permet pas de représenter le processus assez précisément. Une abstraction trop faible ou une résolution trop élevée correspond à représenter des éléments ou des modifications qui ne sont pas nécessaires. Le choix du niveau d'abstraction est donc primordial pour représenter un système de manière adéquate en fonction de ce que l'on veut en faire.

La représentation de processus sous forme de système est utilisée dans le domaine de l'automatique pour contrôler (dans le sens gérer et non pas vérifier) le comportement d'un processus. Un contrôleur est intégré dans une boucle de rétroaction pour amener les sorties vers des valeurs de référence (Fig. 11).

Les processus de création et d'encodage des images sont intrinsèquement discrets. Les images sont traitées les unes après les autres de manière séquentielle. De ce fait, nous utilisons une représentation discrète, où k correspond à un instant ou à l'indice de chaque nouvelle image.

3.2.1 Encodeur et transmission

Les processus d'encodage et de transmission sont représentés sous forme de systèmes. L'encodeur reçoit le flux d'images sources pour en faire un flux d'images compressées. Cependant, pour l'approche système, cela ne correspond pas aux entrées et sorties. Les entrées

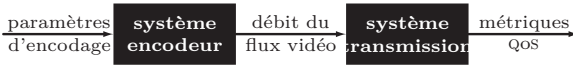


Fig. 12: Les processus d'encodage et de transmission sont représentés sous forme de systèmes.

du système encodeur sont les paramètres d'encodage sélectionnés dans le Chap. 2 : la fréquence f et la quantification q . Le débit du flux vidéo encodé fait office de sortie du système encodeur. De même, l'entrée du système transmission représente le débit de transmission, c'est-à-dire le débit du flux vidéo. Les métriques de la qualité de service (QoS pour *Quality of Service*) de la transmission, comme le taux de perte, le délai et le gigue forment les sorties du système transmission. Etant donné que la sortie du système encodeur et l'entrée du système transmission sont identiques, les deux systèmes peuvent être mis en série (Fig. 12). Les systèmes peuvent être ainsi vus comme un nouveau système plus général avec les paramètres d'encodage comme entrées, et les métriques QoS.

3.2.2 Contrôle de la transmission

Le canal de communication transfère des données entre deux éléments. Nous caractérisons la capacité de transmission par le *débit disponible* pour une application. Ce débit disponible peut varier en fonction de la performance de l'infrastructure de transmission ou du trafic de données en cours. Le débit d'envoi du flux vidéo d'AR est fonction du débit disponible pour obtenir une transmission optimale.

En utilisant l'approche système proposée, il est possible de mettre en place un mécanisme de contrôle pour la transmission qui détermine le *débit de référence* de notre système transmission en fonction des métriques de la QoS. Nous faisons l'hypothèse que ce mécanisme existe et donne en tout temps l'information sur le débit de référence. Il est possible d'utiliser des mécanismes existants qui sont généralement propres à un type de réseau [50–54].

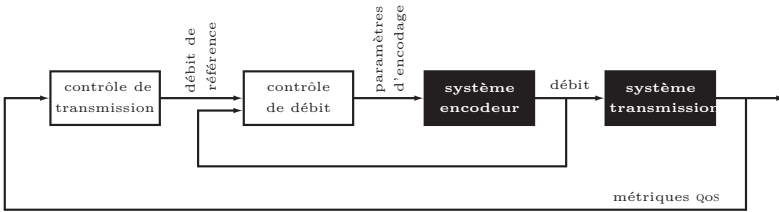


Fig. 13: La boucle externe détermine le débit disponible. Le contrôle de débit de la boucle interne contrôle le débit du flux vidéo.

Nous représentons ce mécanisme par un contrôleur de transmission bouclé sur le système transmission.

De manière similaire, le contrôle de débit est représenté par un contrôleur d'encodeur bouclé sur le système encodeur. Comme les deux systèmes peuvent être mis en série, nous obtenons un contrôle en cascade, où la boucle externe donne le débit de référence, calculé à partir du débit disponible, comme référence à la boucle interne du contrôle de débit (Fig. 13).

3.2.3 Modèle du système transmission

Un modèle du système transmission est nécessaire pour évaluer l'impact du transfert du flux vidéo sur la QoE. Des nombreux paramètres de la QoS ayant de l'influence [41, 55–57], nous ne retenons que celui qui est lié à la métrique QoE : le délai de transmission. En effet, le délai de transmission constitue une partie importante du délai global pris en compte dans le modèle de QoE. De plus, le délai de transmission peut varier fortement, alors que les autres délais (d'acquisition, d'encodage et d'affichage) sont généralement fixes.

Nous proposons un modèle simplifié pour la transmission qui consiste en un tampon de type premier entré premier sorti (FIFO pour *First In First Out*) de taille infini. Il se vide grâce au débit disponible variable. Le débit du flux vidéo est l'entrée, et le délai la sortie

du système. Le débit disponible est vu comme une perturbation. Si le débit du flux vidéo est inférieur ou égal au débit disponible, chaque image du flux vidéo est transmise directement. Si le débit du flux vidéo est supérieur au débit disponible, les images sont stockées temporairement dans le tampon. Le tampon se vide à un certain débit. Les images subissent ainsi un délai, mais ne sont pas perdues.

Le *délai de transmission* d_T correspond à la période entre la création de l'image au temps k^* et sa sortie du canal de communication $k^* + d_T$. Ce délai dépend de son niveau V dans le tampon de transmission à l'instant k^* et du débit disponible r_a , supposé constant de l'instant k^* à l'instant $k^* + d_T$. Comme le tampon est de type FIFO, cette image ne subit pas de délai dû aux images qui arrivent après. Ainsi, le débit de remplissage n'influence plus le délai à partir du temps $k = k^*$.

$$d_T = \frac{V(k^*)}{r_a} \quad (8)$$

Le niveau du tampon $V(k^*)$ est calculé sur le principe de remplissage et vidage d'une cuve représentant le tampon infini (Fig. 14). Le niveau (volume) V du tampon dépend des débits de remplissage $r(k)$ et de vidage $r_a(k)$. En connaissant le niveau initial $V(0)$, que nous supposons nul, ainsi que l'évolution des débits $r(k)$ et $r_a(k)$, il est possible de calculer le niveau $V(k^*)$ où l'image est créée.

Définition 6 (Modèle du système transmission). *Le système transmission est modélisé par un tampon FIFO. Il n'y a pas de pertes de données lors de la transmission. Cependant, les images subissent un délai d_T lorsque le débit r excède le débit disponible r_a , conformément à l'Equ. (8).*

La valeur du débit de référence r_r que le contrôle de transmission donne au contrôle de débit est généralement lié au débit disponible r_a . Cette valeur dépend de l'algorithme de contrôle de transmission qui n'est pas traité dans cette thèse.

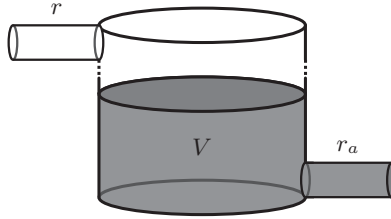


Fig. 14: Le système de transmission est modélisé par un tampon infini représenté par une cuve.

3.3 Modèle du système encodeur

La synthèse du contrôleur nécessite le modèle du système à contrôler. Nous proposons un modèle pour le système encodeur, dont les entrées sont les paramètres d'encodage f et q et la sortie le débit du flux vidéo r . Le système est modélisé de manière générique avec un niveau d'abstraction adapté pour le contrôle.

Cette section introduit la représentation réelle non linéaire du système encodeur, puis présente le modèle linéaire abstrait.

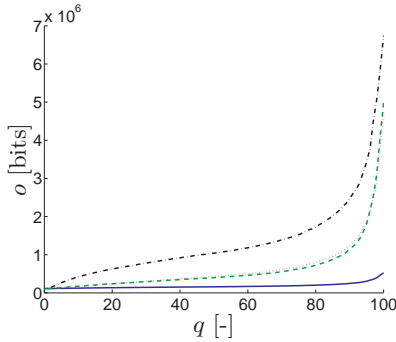
3.3.1 Représentation non linéaire

Le débit est défini comme une quantité par unité de temps. Le débit d'un flux vidéo r correspond au rapport entre la taille de l'image compressée o et la période de temps p entre deux images successives.

La taille de l'image compressée dépend du paramètre de quantification q et du contenu de l'image. La Fig. 15 montre la relation entre le paramètre de quantification et la taille de l'image compressée, où le paramètre de quantification correspond au facteur de compression JPEG de la librairie présentée à la Sect. 4.1.7. Le paramètre de quantification influence la taille de l'image compressée de manière non linéaire. Il est difficile de caractériser l'impact de la valeur du paramètre de quantification et du contenu sur la taille de l'image compressée.



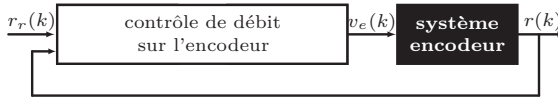
(a) Quatre images de référence (Annexe C) avec un contenu de plus en plus riche.



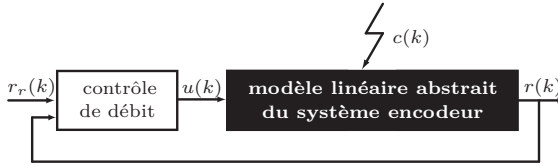
(b) Taille o en fonction de la quantification q pour les quatre images ; dans l'ordre bleu continu, vert traitillé, rouge pointillé et noir mixte.

Fig. 15: Relation non linéaire entre la quantification q (ici le facteur de compression JPEG) et la taille de l'image compressée o .

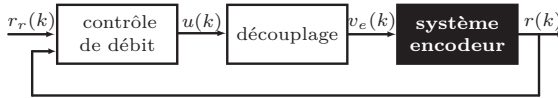
L'algorithme de contrôle de débit adapte les paramètres d'encodage pour que le débit r suive le débit de référence r_r ; quelque soit la variation du contenu du flux vidéo (Fig. 16a). Les valeurs des paramètres d'encodage sont calculées pour chaque image d'indice k . Deux paramètres d'encodage sont pris en compte, la fréquence f et la quantification q , sélectionnés au Chap. 2. Le système encodeur a deux entrées et exhibe un comportement non linéaire entre la quantification q et le débit r .



(a) Le contrôle de débit sur l'encodeur adapte directement les paramètres d'encodage. Le système encodeur se comporte de manière non linéaire.



(b) Le contrôleur est conçu à partir du modèle linéaire monovariante du système encodeur, dont l'entrée unique est u .



(c) Le découplage permet de passer de l'entrée unique u issu du contrôle de débit aux paramètres d'encodage v_e de l'encodeur réel.

Fig. 16: Le contrôle de débit adapte les paramètres d'encodage v_e (éventuellement par l'intermédiaire de l'entrée unique u) pour que le débit r suive la référence r_r .

L'étude et le contrôle des systèmes non linéaires est un domaine complexe. Afin de pouvoir utiliser une plus large palette de méthodes, nous proposons une représentation linéaire du système encodeur.

3.3.2 Modèle linéaire

Plutôt que d'utiliser la représentation non linéaire (relation entre les paramètres d'encodage et le débit) de l'encodeur, nous proposons un modèle abstrait linéaire. La linéarité concerne le lien entre une nouvelle entrée abstraite (liée au paramètres d'encodage) et le débit. Il ne s'agit pas d'une linéarisation classique de système, mais juste d'une représentation arbitraire qui permet d'utiliser des outils linéaires.

Le débit $r(k)$ est la sortie du système encodeur, c'est-à-dire le débit correspondant à l'instant k . Les entrées du système encodeur sont la fréquence $f(k)$ et la quantification $q(k)$. Le contenu est vu comme une perturbation car il ne peut être contrôlé. Nous utilisons la métrique $c(k)$ pour quantifier l'impact du contenu sur le débit. Cette métrique de contenu est directement liée au contenu de l'image et peut donc être différente pour chaque image. Le débit est supposé proportionnel à la métrique de contenu. Cette hypothèse sera justifiée dans le Chap. 4. Ainsi, le modèle intrinsèquement discret de l'encodeur est donné par :

$$r(k) = g(f, q, k)c(k) \quad (9)$$

où g est la fonction qui donne les combinaisons réalisables de paramètres d'encodages.

Définition 7 (Fonction réalisable dans l'espace d'encodage).

La fonction g est une fonction implicite non linéaire paramétrée par l'entrée unique u .

$$g(f, q, k) = u(k) \quad (10)$$

où $u(k)$ est la nouvelle entrée unique au temps k . La fonction réalisable g couvre toutes les combinaisons de paramètres d'encodage $(f(k), q(k))$ qui correspondent à l'entrée unique $u(k)$ donnée (Fig. 17). La fonction g dépend du contenu de l'image et évolue dynamiquement ; elle sera déterminée à l'aide de la méthode du mapping (mise en correspondance) (Chap. 4).

Dans cette section, la fonction g permet de définir un modèle de l'encodeur indépendant des paramètres d'encodage. À partir des Equ. (9) et (10), nous obtenons un modèle pour le contrôle.

Définition 8 (Modèle abstrait de l'encodeur). *Le modèle abstrait de l'encodeur est discret à une entrée et une sortie (SISO pour Single Input Single Output).*

$$r(k) = u(k)c(k) \quad (11)$$

Ce modèle possède une entrée $u(k)$, une sortie $r(k)$ et une perturbation $c(k)$ sans mention des paramètres d'encodage (Fig. 16b).

Pour l'implémentation, la combinaison des valeurs des paramètres d'encodage $f(k)$ et $q(k)$ est choisie à partir de l'entrée unique $u(k)$ et du bloc de découplage (Fig. 16c). Le découplage utilise la méthode du *mapping* présentée dans le Chap. 4.

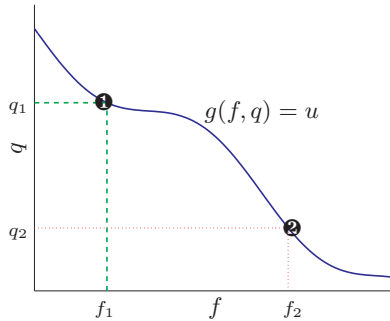


Fig. 17: La fonction $g = u$ (bleu continu) passe par toutes les combinaisons (f, q) possibles. Pour une valeur de l'entrée unique u donnée, n'importe quelles de ses combinaisons (f, q) peuvent choisies, comme par exemple ① et ②. La fonction g est déterminée à l'aide de la méthode du *mapping* présentée au Chap. 4.

3.4 Conclusion

Dans ce chapitre, nous avons développé une représentation entrée/sortie avec un actuateur linéaire pour le contrôle de débit. Cela permet de suivre une consigne de débit, malgré un contenu d'images variable. Nous avons utilisé l'approche système qui permet d'obtenir une représentation adaptée au contrôle. Nous avons modélisé la transmission par un tampon FIFO, où les données subissent un délai si le débit dépasse le débit disponible. Le débit est lié de manière non linéaire à certains paramètres d'encodage. Afin de pouvoir développer des stratégies de contrôle linéaire, nous avons développé un modèle linéaire de l'encodeur où le débit est proportionnel à une entrée abstraite. Cela permet de traiter plusieurs paramètres d'encodage de manière abstraite.

Le mécanisme de contrôle de débit proposé est situé au niveau de l'encodage. Ceci permet de supprimer le tampon de sortie du flux vidéo et de diminuer le délai relatif. Le modèle abstrait proposé combiné à la méthode du *mapping* (Chap. 4) permet de maximiser la qualité tout en contrôlant le débit, comme proposé précédemment pour des applications connexes [45]. L'utilisation du modèle abstrait proposé permet en plus d'intégrer cette notion de qualité indépendamment de l'algorithme de compression choisi. Par rapport au contrôle de débit classique, les stratégies de contrôle basées sur l'automatique (Sect. 3.1.2) permettent d'améliorer les performances. Cependant elles ne garantissent pas la stabilité car le modèle utilisé n'est pas linéaire [49]. Le modèle linéaire proposé permet de concevoir un contrôleur en tenant compte des performances et de la stabilité.

Le contrôle de débit développé modifie la valeur de la variable abstraite d'entrée unique $u(k)$. Pour être implémenté en pratique, il est nécessaire de déterminer la valeur des paramètres d'encodage (fréquence et quantification) sur la base de cette variable d'entrée, grâce à la fonction réalisable g . Le Chap. 4 présente la méthode qui permet d'y parvenir.

Le contrôle de débit développé dans ce chapitre est utilisé par le schéma d'adaptation présenté dans le Chap. 5. Grâce à l'intégration du modèle de QoE présenté dans le Chap. 2 et le *mapping* présenté dans le Chap. 4, les paramètres d'encodage sont adaptés pour obtenir la meilleure expérience utilisateur possible.

Méthode du mapping

Le mapping permet d'appliquer le contrôle de débit abstrait à un encodeur réel.

Le contrôle de débit présenté dans le Chap. 3 permet de déterminer la valeur de l'entrée unique u pour suivre le débit de référence du flux vidéo. Pour que le contrôle de débit soit applicable, il est nécessaire de transposer cette entrée unique en paramètres d'encodage. Les paramètres d'encodage choisis (fréquence et quantification) et l'entrée unique u sont reliés par la fonction réalisable g dans l'espace d'encodage.

Ce chapitre présente une méthode générique pour déterminer la fonction g d'un encodeur. Ainsi, la valeur des paramètres d'encodage est choisie en fonction de la valeur de l'entrée unique (Fig. 18). Cela permet d'appliquer le contrôle de débit de manière générique.

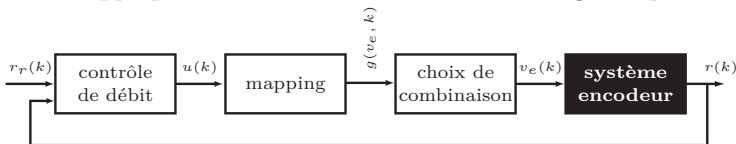


Fig. 18: Le contrôle de débit donne la valeur de l'entrée unique $u(k)$. La méthode du *mapping* détermine la fonction réalisable g qui permet de choisir les paramètres d'encodage $v_e(k)$.

La fonction réalisable g dans l'espace d'encodage est équivalente à une nouvelle fonction h dans un autre espace. Nous construisons cette nouvelle fonction grâce à la connaissance du système encodeur. Un *mapping* permet ensuite de faire correspondre ces deux fonctions et ainsi de déterminer la fonction réalisable dans l'espace d'encodage.

Dans ce chapitre, la méthode du *mapping* est appliquée à la classe d'encodeurs de type transformée en cosinus discrète (DCT pour *Discrete Cosine Transform*). La Sect. 4.1 rappelle le principe d'encodage DCT. Les étapes de la compression d'une image en JPEG (Joint Photographic Experts Group) y sont présentées. La Sect. 4.2 introduit la fonction réalisable h dans l'espace linéaire. Le *mapping* Φ , permettant de passer entre l'espace linéaire de h et l'espace d'encodage de g , est présenté dans la Sect. 4.3. L'analyse dans le ρ -domaine de la Sect. 4.4 présente une nouvelle variable utilisée par le *mapping* Φ_2 . L'algorithme permettant de déterminer le *mapping* en temps réel est présenté dans la Sect. 4.5. La Sect. 4.6 présente l'application du contrôle de débit à un flux vidéo encodé en JPEG. La Sect. 4.7 conclut ce chapitre en rappelant les principales contributions.

4.1 Compression DCT

L'encodage vidéo a pour objectif de diminuer le débit du flux vidéo tout en gardant une qualité demandée. L'encodeur-décodeur (CODEC pour *COder-DECoder*) est le mécanisme responsable de la compression et de la décompression du flux vidéo. La compression vidéo tire partie des redondances spatiales et/ou temporelles du flux d'images pour diminuer le volume d'information nécessaire pour représenter le flux vidéo.

Nous expliquons l'encodage vidéo en traitant le cas simple de la compression M-JPEG (Motion Joint Photographic Experts Group). Chaque image est compressée séparément en JPEG et seule la redondance spatiale est traitée. Il est ainsi possible d'illustrer l'encodage vidéo avec la compression d'une seule image. L'algorithme de

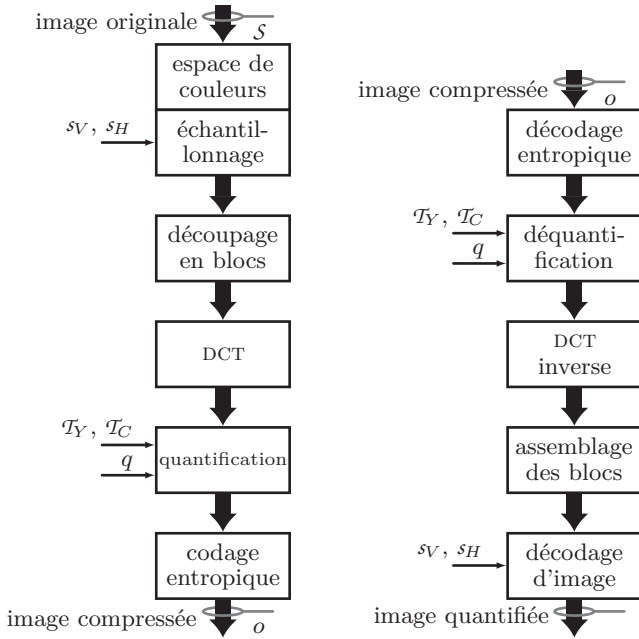


Fig. 19: Etapes d'encodage et de décodage d'une image pour un encodeur DCT. La taille de l'image o compressée dépend du facteur de quantification q , des tables de quantification \mathcal{T} et des facteurs d'échantillonnage s .

compression JPEG transforme une image en un volume de données compact précédé d'un en-tête¹.

Les étapes de la procédure d'encodage JPEG sont illustrées avec une image de référence (800×600 pixels avec une profondeur de couleurs de 24 bits en RGB (Red Green Blue) avec une résolution de 8 bits). La Fig. 19 montre les étapes de la procédure d'encodage.

1. *header*

La taille S de l'image non compressée dépend de sa largeur W , sa hauteur H et de la profondeur de couleurs D .

$$S = WHD \quad (12)$$

4.1.1 Espace de couleurs et échantillonnage

L'image de base est représentée dans l'espace de couleurs YCbCr (luminance-chrominance). L'image est composée de trois composantes :

1. la luminance
2. la chrominance bleue
3. la chrominance rouge.

Cet espace de couleurs permet de réduire la taille d'une image grâce au fait que l'oeil humain est plus sensible à la luminance qu'à la chrominance. Ainsi, il est possible de diminuer la résolution de la chrominance sans altérer la qualité de l'image. Le sous-échantillonnage consiste à réduire la taille des composantes de chrominance en diminuant la résolution verticale et/ou horizontale. Après cette étape, les trois composantes sont traitées de manière similaire.

La taille de l'image sous-échantillonnée S' dépend des facteurs d'échantillonnage vertical s_V et horizontal s_H . Le sous-échantillonnage s est généralement de $\frac{1}{2}$ ou $\frac{1}{4}$. Il se peut que la hauteur/largeur de l'image ne soit pas divisible par 2 ou 4. Dans ce cas, la dimension de l'image est artificiellement augmentée en ajoutant des lignes ou colonnes. La taille de la composante de luminance est notée S'_Y , celle de la chrominance S'_C . La luminance et les chrominances ont chacune une résolution de 8 bits, notée d .

$$S' = S'_Y + 2S'_C \quad (13)$$

$$S'_C = \lceil Ws_H \rceil \lceil Hs_V \rceil d \quad (14)$$

$$S'_Y = WHd \quad (15)$$

La Fig. 20 illustre cette étape de la procédure de compression.

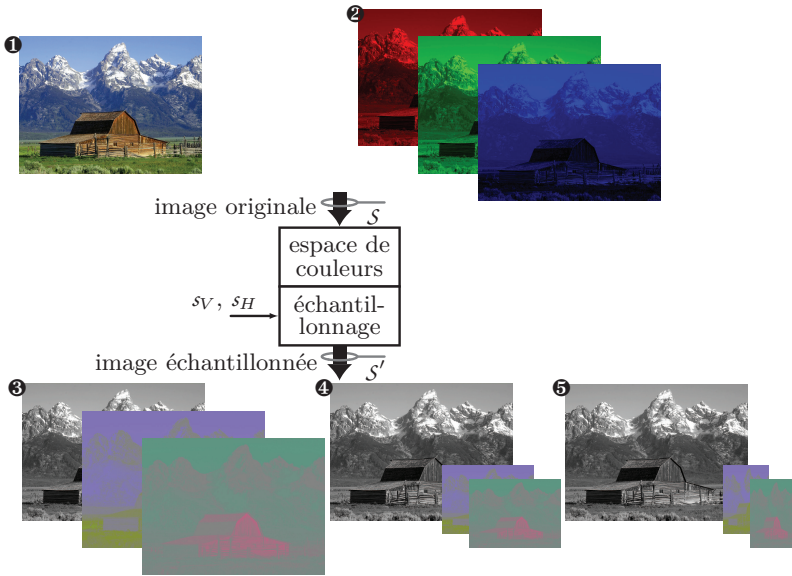


Fig. 20: Transformation d'espace de couleur et sous-échantillonnage. L'image initiale ❶ est représentée en RGB ❷, puis en YCbCr avec différents facteurs échantillonnage : ❸ $s_H = s_V = 1$, ❹ $s_H = s_V = \frac{1}{2}$ et ❺ $s_H = \frac{1}{4}$, $s_V = \frac{1}{2}$.

4.1.2 Découpage en blocs

L'algorithme JPEG découpe les trois composantes de l'image en blocs, généralement de 8×8 pixels. Chaque bloc f contient 64 pixels qui ont en principe une grande ressemblance.

Le nombre de blocs N est composé du nombre de blocs de luminance N_Y et du nombre de blocs de chaque chrominance N_C . A nouveau, la dimension de l'image est augmentée artificiellement si les valeurs ne sont pas divisibles.

$$N = N_Y + 2N_C \quad (16)$$

$$N_C = \left\lceil \frac{H_{sV}}{8} \right\rceil \left\lceil \frac{W_{sH}}{8} \right\rceil \quad (17)$$

$$N_Y = \left\lceil \frac{H}{8} \right\rceil \left\lceil \frac{W}{8} \right\rceil \quad (18)$$

La Fig. 21 illustre cette étape de la procédure de compression.

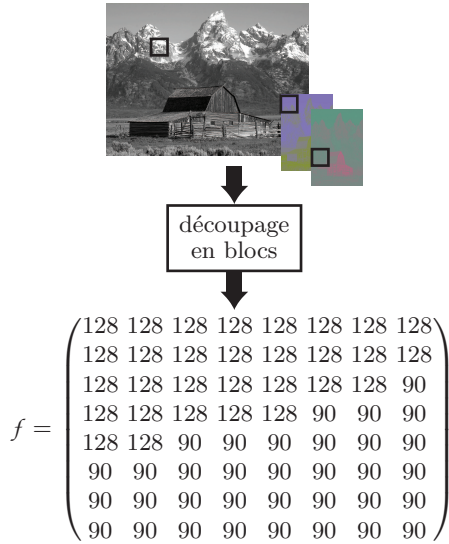


Fig. 21: Découpage des trois composantes en blocs f de pixels. L'un des N blocs est illustré.

4.1.3 DCT

La méthode de la DCT permet de transcrire des données du domaine d'amplitude vers le domaine fréquentiel. Dans le cas de la compression JPEG, chaque bloc de 8×8 pixels est représenté dans le domaine fréquentiel par une matrice de 8×8 coefficients DCT. La DCT bi-dimensionnelle de type II est utilisée pour convertir les pixels $f(x, y)$ en fréquences $\mathcal{F}(u, v)$. La transformée et son inverse s'expriment mathématiquement par [58] :

$$\mathcal{F}(u, v) = \frac{1}{4} C(u) C(v) \left[\sum_{x=0}^7 \sum_{y=0}^7 f(x, y) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \right] \quad (19)$$

$$f(x, y) = \frac{1}{4} \left[\sum_{u=0}^7 \sum_{v=0}^7 C(u) C(v) \mathcal{F}(u, v) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \right] \quad (20)$$

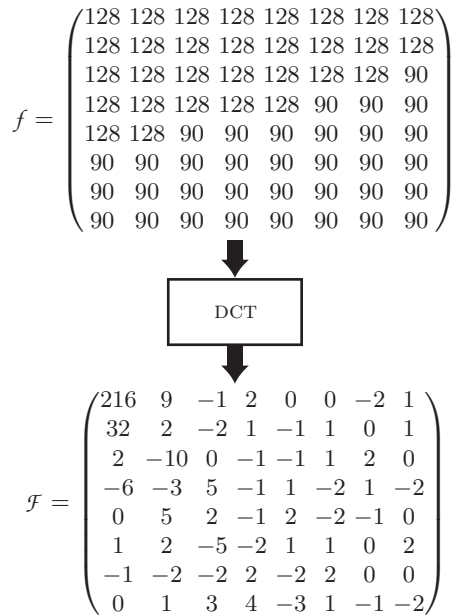
avec les coefficients normalisant :

$$\begin{cases} C(u), C(v) = \frac{1}{\sqrt{2}} & \text{pour } u, v = 0 \\ C(u), C(v) = 1 & \text{sinon.} \end{cases} \quad (21)$$

Le nombre de matrices de coefficients \mathcal{F} correspond au nombre de blocs N . La DCT ne diminue pas la quantité de données nécessaire pour représenter l'image. Cependant, elle permet de séparer les fréquences entre elles et ainsi de supprimer les fréquences les plus élevées afin de réduire la quantité de données.

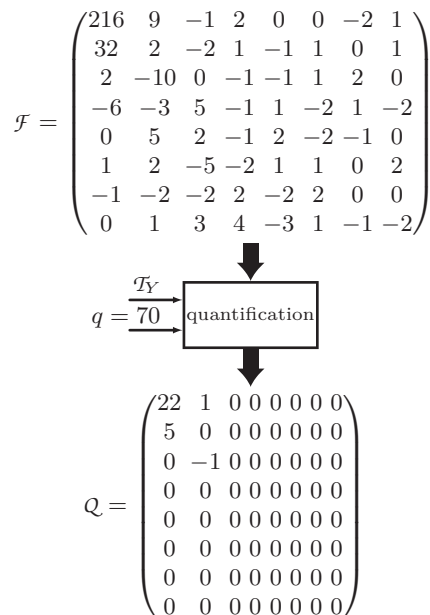
La DCT est l'étape qui demande le plus de calculs – et donc de temps – dans la compression JPEG.

La Fig. 22 illustre cette étape de la procédure de compression.

Fig. 22: DCT sur la matrice f .

4.1.4 Quantification

La quantification permet de diminuer la taille de l'image compressée. Cette étape réduit également la qualité de l'image compressée. L'oeil humain est capable de remarquer des différences de contraste entre des zones éloignées, mais il est moins à même de distinguer des différences entre des zones rapprochées et donc à fréquence élevée. La quantification utilise ce principe pour réduire les composantes à hautes fréquences des matrices \mathcal{F} de coefficients DCT. Les coefficients DCT sont divisés par des coefficients de quantification. Plus la fréquence est élevée, plus le coefficient de quantification sera grand. Il y a donc plus de chances qu'un coefficient lié à une

Fig. 23: Quantification d'une matrice \mathcal{F} de luminance.

Plus le facteur \mathcal{W} est grand (ou q petit), plus il y aura de coefficients DCT quantifiés nuls. Et grâce au codage entropique, la taille de l'image compressée est liée au nombre de coefficients non-nuls.

La Fig. 23 illustre cette étape de la procédure de compression.

4.1.5 Codage entropique

Le codage entropique est un algorithme de codage sans perte qui permet de représenter une suite de signes à dimension fixe (par exemple des nombres) par une suite de signes à dimension variable (par exemple des codes binaires). Le codage de Huffman est l'algo-

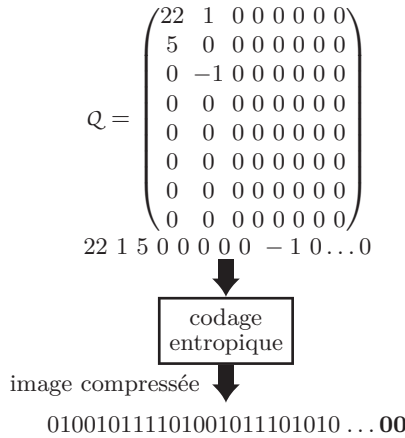


Fig. 24: Codage entropique en zigzag pour la matrice Q .

rithme utilisé pour la compression JPEG. La suite binaire est généralement plus compacte que la suite de nombres. Par exemple, si la suite de signes en entrée se termine par de nombreux zéros, un seul signe la remplacera avantageusement.

Afin de maximiser la probabilité d’avoir de nombreux zéros en fin de suite, la matrice des coefficients quantifiés est lue en zigzag. Les coefficients liés aux fréquences les plus élevées ayant été annulés se retrouvent en fin de lecture.

La Fig. 24 illustre cette étape de la procédure de compression.

4.1.6 Décompression JPEG

La décompression de l’image utilise le procédé inverse de la compression (Fig. 19). Certaines informations nécessaires à la décompression sont contenues dans l’en-tête de l’image. L’image compressée subit le décodage entropique pour retrouver la matrice des coefficients DCT quantifiés Q . Cette matrice est multipliée par le facteur

de pondération \mathcal{W} et par la table de quantification \mathcal{T} . Puis la DCT inverse est appliquée à la matrice de coefficients \mathcal{F} pour obtenir les blocs f de 8×8 pixels. La qualité de l'image quantifiée est évidemment plus faible que la qualité de l'image originale.

4.1.7 Taille de l'image compressée

La taille de l'image compressée correspond à la quantité de données nécessaire pour représenter la suite de signes après l'encodage entropique. Elle dépend principalement du facteur de quantification q et du contenu de l'image.

Dans cette thèse, les images sont encodées en JPEG sans sous-échantillonnage avec la librairie distribuée par le "Independent JPEG Group" (<http://www.ijg.org/>).

4.2 Fonction réalisable dans l'espace linéaire

Nous cherchons à déterminer la fonction réalisable $g(f, q, k)$ dans l'espace d'encodage en fonction du paramètre u . Cette section présente une nouvelle fonction réalisable h équivalente à g , dans un espace où le paramètre u a un sens naturel. Une seconde approche de modélisation du débit permet de définir h .

Le débit r correspond au volume de données par unité de temps. Pour le débit d'un flux vidéo, le volume de données correspond à la taille de l'image compressée o (ou plus généralement au volume de données nécessaire pour reconstituer une nouvelle image), alors que l'unité de temps est donnée par la période de temps entre deux images successives p . Ainsi, le débit à l'instant k est mesuré grâce à la taille compressée de l'image k et la période de temps jusqu'à la prochaine image $k + 1$.

$$r(k) = \frac{o(k)}{p(k)} \quad (26)$$

Le facteur de quantification q n'est pas proportionnel à la taille de l'image compressée o (Fig. 15). Pour les besoins du développement, nous introduisons une nouvelle variable linéaire l de telle sorte qu'elle soit proportionnelle à la taille de l'image compressée o . Cette proposition sera confirmée dans la Sect. 4.4.1, la variable linéaire l correspondant à une métrique interne au processus de compression. Le facteur de multiplication entre la variable linéaire et la taille de l'image correspond à la métrique de contenu $c(k)$.

$$o(k) = l(k)c(k) \quad (27)$$

A partir des Equ. (26) et (27), le débit devient ainsi fonction de la variable linéaire et de la période de temps.

$$r(k) = \frac{o(k)}{p(k)} = \frac{l(k)c(k)}{p(k)} \quad (28)$$

Nous définissons h comme étant la fonction de la variable linéaire et de la période de temps pour obtenir une nouvelle expression pour le débit.

$$h(p, l, k) = \frac{l(k)}{p(k)} \quad (29)$$

$$r(k) = h(p, l, k)c(k) \quad (30)$$

Pour rappel, le modèle linéaire de l'encodeur est donné par l'Equ. (11) :

$$r(k) = u(k)c(k)$$

Ainsi, les Equ. (11), (29) et (30) nous donnent :

$$h(p, l, k) = \frac{l(k)}{p(k)} = u(k) \quad (31)$$

$$l(k) = p(k)u(k) \quad (32)$$

La fonction $h(p, l, k)$ est donc paramétrée par u , où le paramètre u peut être interprété comme la pente d'une droite appartenant au faisceau de droites dans le plan p et l (Fig. 25).

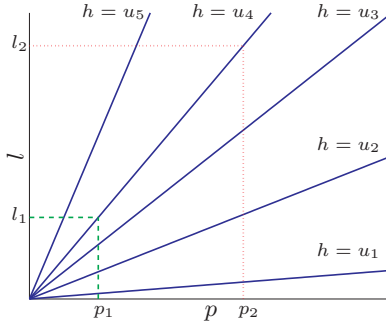


Fig. 25: La fonction réalisable dans l'espace linéaire $h = u$ correspond à une droite. Pour une valeur de l'entrée unique u donnée, n'importe lequel de ses points (p, l) peut être choisi, comme par exemple (p_1, l_1) ou (p_2, l_2) pour $h = u_4$.

Nous pouvons définir la fonction h et son espace de manière similaire à la définition de la fonction g (Déf. 7).

Définition 9 (Espace linéaire). *L'espace linéaire est l'espace engendré par la variable linéaire l et la période p . Cet espace est appelé "linéaire" car un débit donné est représenté par une droite.*

Définition 10 (Fonction réalisable dans l'espace linéaire). *La fonction h est une fonction implicite non linéaire paramétrée par u .*

$$h(p, l, k) = u(k) \quad (33)$$

où $u(k)$ est l'entrée unique au temps k . La fonction réalisable h couvre toutes les combinaisons de paramètres d'encodage $(l(k), p(k))$ qui correspondent à l'entrée unique $u(k)$ donnée (Fig. 25).

4.3 Définition du mapping

Les fonctions $g(f, q, k)$ (Déf. 7 à la Sect. 3.3.2) et $h(p, l, k)$ (Déf. 10) sont toutes deux paramétrées par u . Ces deux fonctions g et h sont équivalentes dans des espaces différents. Cette section présente le *mapping* qui permet de faire correspondre g et h .

$$g(f, q, k) = h(p, l, k) = u(k) \quad (34)$$

Définition 11 (Mapping Φ). *Le mapping Φ est l'application qui fait correspondre l'espace linéaire (engendré par p et l) à l'espace d'encodage (engendré par f et q). Ce mapping permet de passer de la fonction h à la fonction g pour déterminer les entrées de l'encodeur f et q . Le mapping Φ et son inverse Φ^{-1} dépendent du contenu de l'image et donc du temps.*

$$(f, q, k) = \Phi(p, l, k) \quad (35)$$

$$(p, l, k) = \Phi^{-1}(f, q, k) \quad (36)$$

Le mapping Φ permet de faire correspondre un point ou une fonction entre les deux espaces (Fig. 26).

Grâce à la connaissance du système, il est possible de dissocier le *mapping* Φ en deux sous-parties indépendantes.

Définition 12 (Mapping Φ_1). *Le mapping Φ_1 est l'application qui fait correspondre la période p à la fréquence f . La fréquence correspond à l'inverse de la période entre deux images successives.*

$$f(k) = \Phi_1(p) = \frac{1}{p(k)} \quad (37)$$

$$p(k) = \Phi_1^{-1}(f) = \frac{1}{f(k)} \quad (38)$$

Le mapping Φ_1 ne dépend pas du temps.

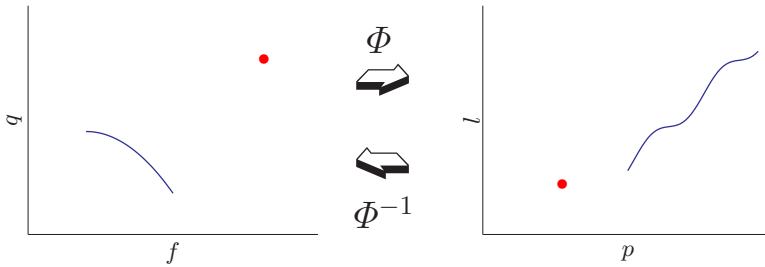


Fig. 26: La mapping Φ fait correspondre un point et une fonction (courbe).

Définition 13 (Mapping Φ_2). *Le mapping Φ_2 est l'application qui fait correspondre la variable linéaire l au facteur de quantification q .*

$$q(k) = \Phi_2(l, k) \tag{39}$$

$$l(k) = \Phi_2^{-1}(q, k) \tag{40}$$

Le mapping Φ_2 dépend du temps.

Le *mapping* Φ permet de définir la fonction g à partir de la fonction h .

$$\begin{aligned} g(f, q, k) &\stackrel{(34)}{=} h(p, l, k) \stackrel{(36)}{=} h(\Phi^{-1}(f, q, k)) \\ &\stackrel{(38)}{=} \stackrel{(40)}{=} h((\Phi_1^{-1}(f), \Phi_2^{-1}(q, k), k) = h\left(\frac{1}{f}, (\Phi_2^{-1}(q, k), k)\right) \\ &\stackrel{(29)}{=} f\Phi_2^{-1}(q, k) \end{aligned} \tag{41}$$

4.3.1 Graphique des quatre quadrants

La Fig. 27 représente graphiquement le *mapping* Φ pour passer de $h(p, l, k)$ à $g(f, q, k)$. Le quadrant en haut à gauche représente la fonction $h(p, l, k)$ dans l'espace linéaire, paramétrée par l'entrée

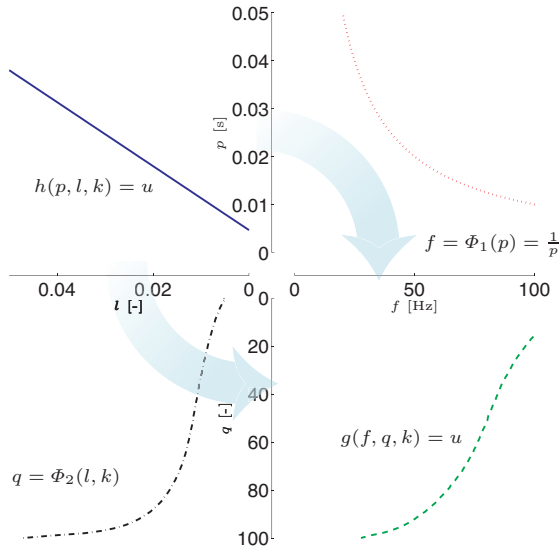


Fig. 27: Le *mapping* Φ fait correspondre la fonction réalisable h (bleu continu) dans l'espace linéaire vers la fonction réalisable g (vert traitillé) dans l'espace d'encodage. Le *mapping* Φ_1 (rouge pointillé) permet de passer de p à f . Le *mapping* Φ_2 (noir mixte) permet de passer de l à q .

unique $u(k)$ sélectionnée par l'algorithme de contrôle de débit. Le quadrant en haut à droite représente le *mapping* Φ_1 qui donne la correspondance statique entre p et f . Le *mapping* Φ_2 entre l et q , illustré dans le quadrant en bas à gauche, dépend du contenu de l'image; il est déterminé grâce à l'algorithme direct (Sect. 4.5). Le *mapping* complet Φ permet de faire correspondre h à g dans l'espace d'encodage (quadrant en bas à droite). La courbe $g(f, q, k)$ correspond à l'ensemble des paramètres d'encodage en adéquation avec l'entrée unique $u(k)$ donnée par le contrôle de débit. L'espace d'en-

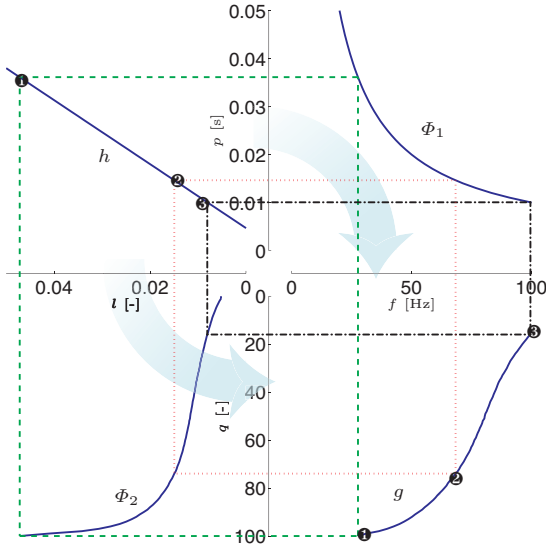


Fig. 28: Présentation du *mapping* de trois points de h vers g . Le point ❶ correspond à un q maximum, le point ❷ à un f maximum et le point ❸ à une valeur intermédiaire.

codage étant l'espace dans lequel le modèle de QoE est défini, il est possible d'utiliser ce modèle pour choisir f et q . Le schéma d'adaptation global (Chap. 5) permet de déterminer les valeurs de f et q qui maximisent la QoE.

La Fig. 28 présente la construction de la fonction g à partir de la fonction h et des *mapping* Φ_1 et Φ_2 . Des points appartenant à h dans l'espace linéaire sont mis en correspondance vers l'espace d'encodage.

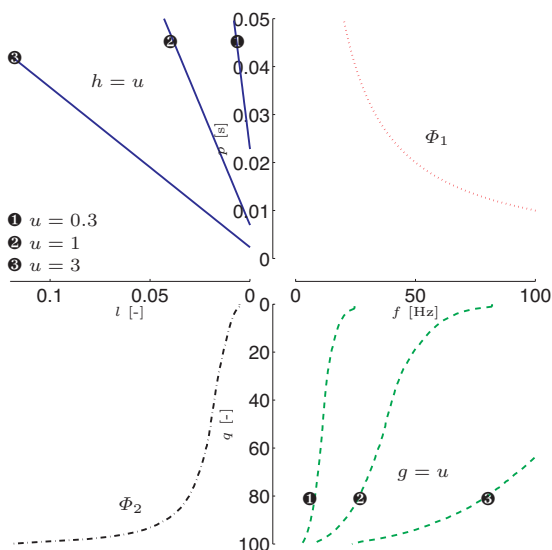


Fig. 29: La fonction g change avec l'entrée unique u . La valeur de l'entrée unique augmente de ❶ à ❸.

Evolution de la fonction réalisable g

La fonction réalisable g évolue en fonction de l'entrée unique u et du mapping Φ_2 qui dépend du contenu du flux vidéo.

La valeur de l'entrée unique u provient du contrôle de débit. Lorsque le débit doit être augmenté, la valeur de u augmente. Ceci a un effet direct sur la fonction réalisable h dont la pente correspond à la valeur de u . Au travers du mapping Φ , ces changements dans h sont transposés sur la fonction réalisable g dans l'espace d'encodage (Fig. 29).

Le mapping Φ_2 fait correspondre la variable linéaire l au paramètre de quantification q . La fonction Φ_2 varie en fonction du

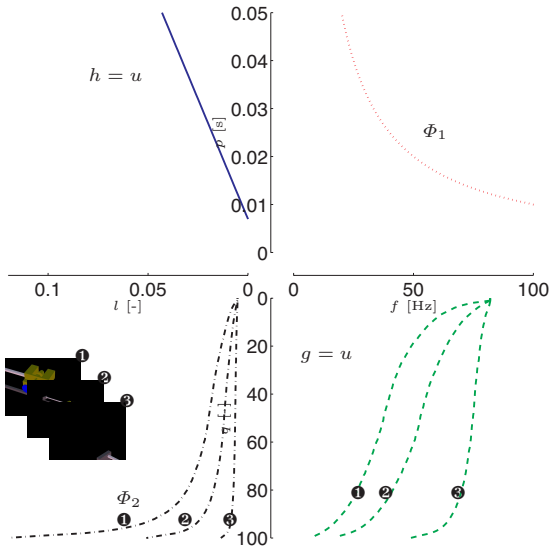


Fig. 30: La fonction g change lorsque le contenu du flux vidéo varie (et donc le *mapping* Φ_2). Le contenu s'enrichit de ❶ à ❸.

contenu du flux vidéo. Ainsi pour une fonction h fixe, le contenu modifie la fonction réalisable g (Fig. 30).

4.4 Mapping Φ_2

Il est nécessaire de déterminer le *mapping* Φ_2 pour définir entièrement la fonction cherchée g et ainsi pouvoir choisir la valeur des paramètres d'encodage $f(k)$ et $q(k)$. Cette section présente la manière de déterminer le *mapping* Φ_2 qui fait correspondre la variable linéaire l au facteur de quantification q pour un encodeur basé sur la DCT. Comme le paramètre de quantification q possède des valeurs

discrètes, il n'est pas possible d'obtenir une relation analytique pour le Φ_2 . Ainsi, le *mapping* Φ_2 est représenté sous forme numérique.

La Sect. 4.4.1 présente l'analyse du ρ -domaine présentée par He et Mitra. Sur la base de cette analyse, la Sect. 4.4.2 justifie le comportement linéaire de la variable l par rapport à la taille de l'image compressée o .

L'algorithme direct permettant de déterminer le *mapping* Φ_2 en temps réel est proposé dans la Sect. 4.5.

4.4.1 Analyse dans le ρ -domaine

La variable linéaire l introduite précédemment correspond à une métrique interne à la compression DCT provenant du modèle linéaire proposé par He et Mitra [59]. Nous présentons ici les résultats obtenus par He et Mitra pour l'analyse dans le ρ -domaine. Dans leur travail, le débit r du flux vidéo est contrôlé uniquement par le paramètre de quantification q , alors que la fréquence f reste constante.

La relation entre q et r est décrite par la fonction débit-quantification dénotée $R(q)$. Si cette fonction est connue, il est possible d'atteindre le débit de consigne r_r avec un paramètre de quantification q correspondant à l'inverse de la fonction $q = R^{-1}(r_r)$. Cependant, la fonction $R(q)$ est complexe et son modèle influence directement la performance du contrôle de débit.

He et Mitra expliquent que les zéros jouent un rôle important dans l'encodage d'images et de flux vidéo. Les auteurs proposent donc d'utiliser une nouvelle variable ρ pour représenter le pourcentage de zéros dans les coefficients DCT quantifiés. Il s'avère que ρ croît de manière monotone avec q . Il existe ainsi un *mapping* bijectif entre ces deux paramètres. Les auteurs montrent que la fonction liant le débit et le pourcentage de zéros $R(\rho)$ est toujours une fonction linéaire.

$$R(\rho) = \theta(1 - \rho) \quad (42)$$

où θ est une constante liée au contenu de l'image.

En résumé, He et Mitra ont montré qu'il existe une variable interne de l'encodeur qui se rapporte au débit de façon linéaire.

4.4.2 Interprétation du ρ -domaine

Notre approche du contrôle de débit et du modèle de l'encodeur se base sur les principes du ρ -domaine. Cela permet d'obtenir l'algorithme direct pour le *mapping* Φ_2 (Sect. 4.5). Nous utilisons la notion de linéarité pour la conception de notre contrôleur ainsi que l'analyse du ρ -domaine pour valider notre modèle linéaire d'encodeur.

La variable linéaire l est définie comme le pourcentage de coefficients DCT non nuls après quantification, c'est-à-dire les coefficients des matrices Q de l'Equ. (22). Cette variable est définie dans le ρ -domaine par $l = (1 - \rho)$. La métrique de contenu c correspond, dans le ρ -domaine, à la constante liée au contenu θ . Cela confirme la relation linéaire pour la taille de l'image compressée o de l'Equ. (27) utilisée pour la fonction réalisable (Sect. 4.2).

Images encodées en JPEG

Des mesures sur quelques images (Annexe C) permettent d'illustrer l'analyse dans le ρ -domaine proposée par He et Mitra pour la compression JPEG ainsi que notre modèle linéaire. La Fig. 31 présente les quatre images utilisées pour illustrer le modèle linéaire de l'encodeur. La Fig. 32 montre la relation linéaire entre la variable l et la taille de l'image compressée o . Il est intéressant de noter que la relation pour la première image est affine et non pas linéaire comme c'est le cas pour les autres images. La prochaine section détaille ce cas.

Relation affine

La première image de la Fig. 31 se différencie des trois autres images par le fait qu'elle a été générée pour la AR. Cette image

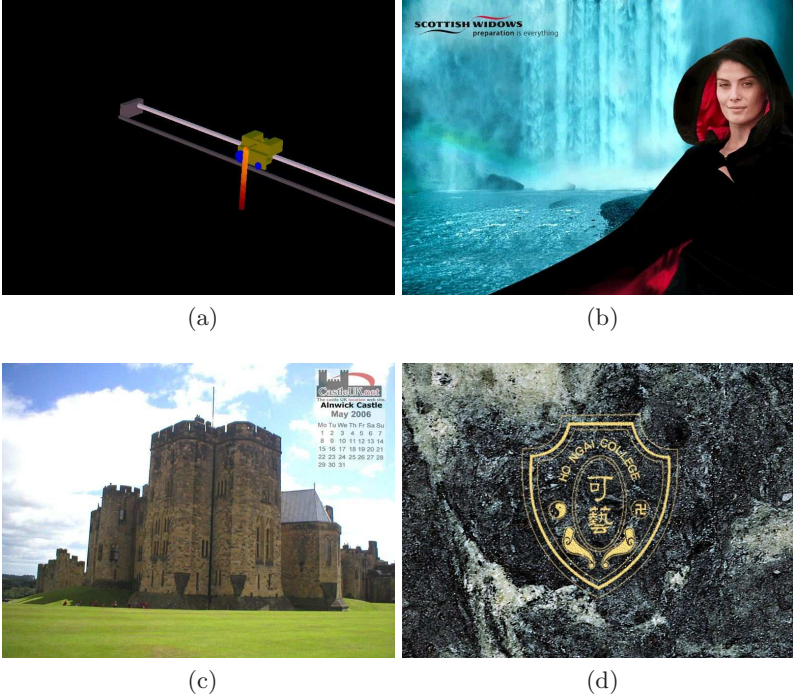


Fig. 31: Images de test (Annexe C) pour le ρ -domaine.

comporte un objet virtuel et un fond uniforme noir (le noir devient transparent lors de l'affichage pour l'utilisateur). Comme on peut le voir sur la Fig. 32a, la relation entre la variable l et la taille de l'image o est affine. Ceci s'explique par le grand nombre de coefficients nuls dans la matrice \mathcal{F} dus au fond noir. De nombreuses matrices pondérées Q sont donc quasiment remplies de zéros, mais le codage entropique ne permet pas de les compresser infiniment.

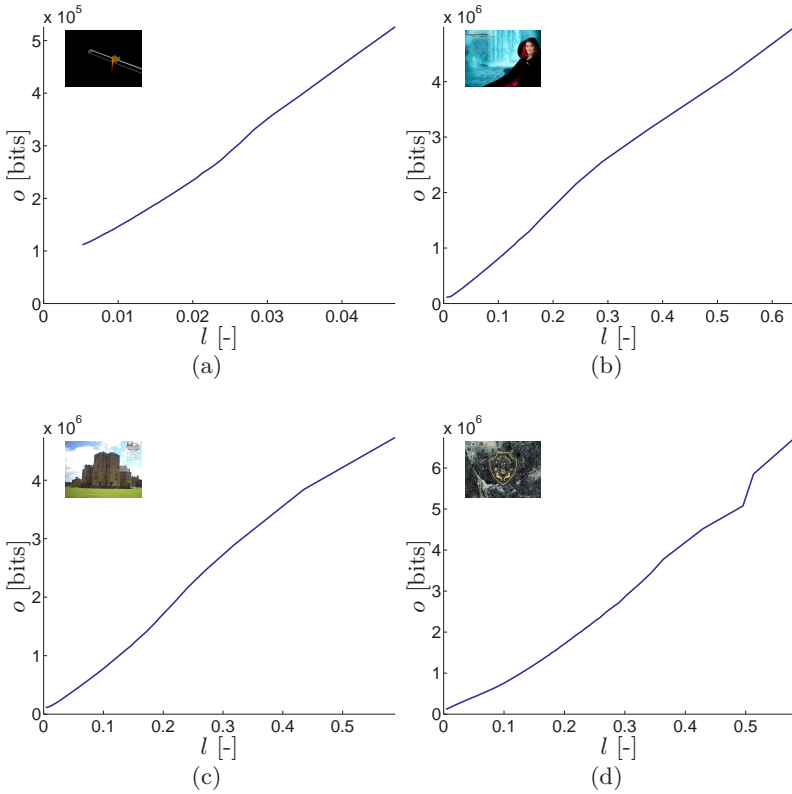


Fig. 32: Relation entre la variable l et la taille de l'image compressée o pour les quatre images de test.

Pour l'implémentation, les équations du modèle doivent être modifiées pour intégrer ce comportement affine. Ainsi, l'Equ. (27) deviendrait :

$$o(k) = (l(k) + \delta(k))c(k) \quad (43)$$

où δ représente le décalage vertical de la relation affine. Sa valeur peut être mesurée pour le besoin de l'implémentation.

Les Equ. (28) et (29) et deviennent respectivement

$$r(k) = \frac{(l(k) + \delta(k))c(k)}{p(k)} \quad (44)$$

$$h(p, l, k) = \frac{(l(k) + \delta(k))}{p(k)} \quad (45)$$

et finalement, l'Equ. 41 définissant g est étendue sans perte de généralité

$$g(f, q, k) = f\Phi_2^{-1}(q, k) + f\delta(k). \quad (46)$$

Pour la suite du chapitre, nous gardons le modèle linéaire validée pour la majorité des images. Cependant les modifications nécessaires sont utilisées lors de l'implémentation lorsque la relation entre la variable l et la taille de l'image compressée o est affine.

4.5 Algorithme direct

Pour pouvoir appliquer le modèle générique de l'encodeur, il est nécessaire de déterminer le *mapping* Φ_2 entre la variable linéaire l et le paramètre de quantification q . Une méthode naïve pour déterminer Φ_2 serait de compresser l'image avec toutes les valeurs possibles de facteurs de quantification, de mesurer le pourcentage de coefficients DCT non nuls et d'en déduire une table de correspondance.

Cette solution à plusieurs passes² n'est pas envisageable car elle nécessite de compresser l'image une infinité de fois afin de déterminer Φ_2 avec précision. L'encodage d'image est un processus qui prend un certain temps, cette solution n'est donc pas envisageable pour une implémentation en temps réel.

Pour permettre un encodage en temps réel, nous proposons un algorithme direct qui ne nécessite qu'une passe³ d'encodage (Fig 33). Après la transformation DCT des matrices f , l'image est représentée sous forme fréquentielle par N matrices \mathcal{F} de 8×8 coefficients. Dans le procédé de compression complet, chaque coefficient $\mathcal{F}(u, v)$ est ensuite pondéré par la valeur $\mathcal{WT}(u, v)$, où \mathcal{W} dépend du facteur de quantification q .

Nous proposons une méthode qui permet de prédire quels coefficients de \mathcal{F} sont annulés lors de l'étape de quantification, sans exécuter la quantification. Nous construisons ainsi le *mapping* Φ_2 numérique en temps réel. A partir des tables de quantification \mathcal{T} , nous déterminons tout d'abord les seuils de quantification, dépendant de q , au-delà desquels les valeurs des coefficients seraient amenées à zéro. Nous comptons ensuite le nombre d'occurrence des valeurs des coefficients que nous regroupons dans un histogramme. Finalement, nous construisons le *mapping* qui relie le pourcentage de coefficients nuls, au paramètre de quantification q grâce aux seuils et aux histogrammes.

4.5.1 Seuils de quantification

Le seuil de quantification correspond à la valeur en dessous de laquelle le coefficient aurait été amené à zéro s'il avait subi l'étape de quantification. Ce seuil dépend du facteur de quantification q et de la valeur de la table de quantification \mathcal{T} pour la position (u, v) .

2. *multi-pass*

3. *single-pass*

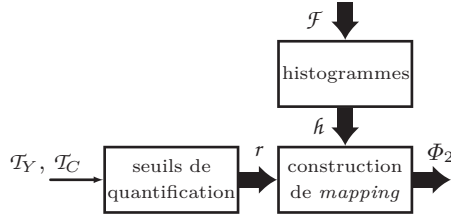


Fig. 33: Etapes de l'algorithme du *mapping* Φ_2 entre l et q .

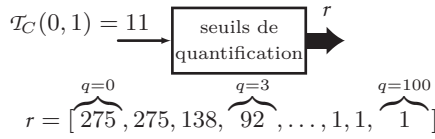


Fig. 34: Etape du calcul des seuils de quantification pour le coefficient de luminance $(u, v) = (0, 1)$.

Les facteurs de quantification possibles sont mis sous la forme d'un vecteur croissant $q = [0, \dots, q, \dots, 100]$. Les seuils sont également représentés sous forme d'un vecteur $r = [r_0, \dots, r_q, \dots, r_{100}]$ où l'indice correspond au facteur de quantification.

$$r_q = \left\lceil \frac{\mathcal{T}(u, v)\mathcal{W}(q)}{2} \right\rceil \tag{47}$$

La Fig. 34 montre comment les seuils r sont calculés pour le coefficient de luminance $(u, v) = (0, 1)$.

Les tables de quantification \mathcal{T} sont fixes et connues à l'avance. De ce fait, le vecteur r peut être calculé à l'avance et simplement utilisé comme tel en temps réel.

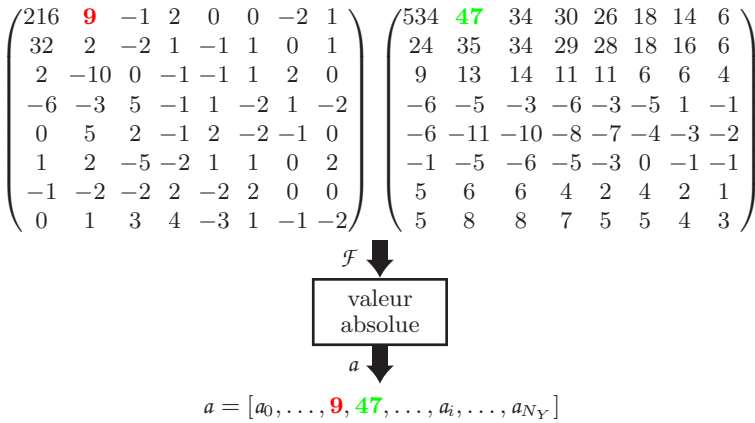


Fig. 35: Deux matrices \mathcal{F} (parmi de nombreuses autres) avec mise en évidence du coefficient $(u, v) = (0, 1)$.

4.5.2 Histogramme des occurrences

Un histogramme des occurrences des valeurs des coefficients est construit pour chaque position (u, v) des coefficients. Nous avons donc $8 \times 8 = 64$ histogrammes pour la composante de luminance. Comme les deux composantes de chrominance utilisent la même table de quantification, nous obtenons 64 histogrammes supplémentaires pour un total de $N_H = 128$ histogrammes.

Pour chaque position (u, v) de coefficients de luminance (idem pour la chrominance), un vecteur a regroupe la valeur absolue du coefficient des N_Y matrices \mathcal{F} . La Fig. 35 montre la création du vecteur a pour la position $(u, v) = (0, 1)$ de la composante de luminance.

Pour chaque position (u, v) de coefficients, un histogramme h est créé à partir du vecteur a . Cet histogramme représente le nombre d'occurrences des valeurs présentes dans le vecteur a . Il est représenté sous forme de vecteur ou graphique (Fig. 36).

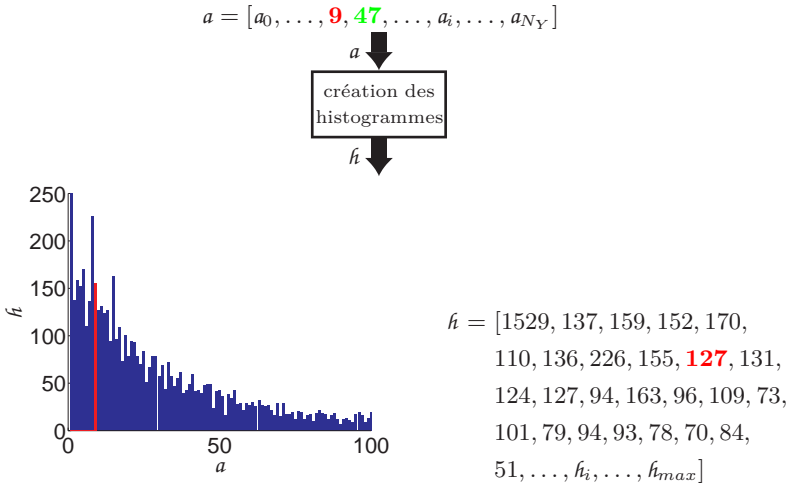


Fig. 36: Histogramme des occurrences pour les valeurs du coefficient $(u, v) = (0, 1)$. La valeur **9** issue de la matrice F de la Fig. 35 incrémente la neuvième occurrence (la valeur **127**, en évidence en rouge sur l’histogramme).

En utilisant les tables de quantification usuelles, le nombre d’histogrammes peut être réduit à environ $N_H = 70$ en regroupant les coefficients qui ont le même facteur dans la table de quantification.

4.5.3 Construction du mapping

Le *mapping* numérique Φ_2 fait correspondre le pourcentage de coefficients non-nuls l au facteur de quantification q . Le pourcentage de coefficients non-nuls correspond au rapport entre le nombre de coefficients qui n’auraient pas été amenés à zéro et le nombre total de coefficients.

Le nombre total de coefficients N_T peut être calculé simplement à partir du nombre de matrices \mathcal{F} .

$$N_T = 64(N_C + N_Y)$$

Grâce aux seuils et aux histogrammes, il est possible de calculer le nombre de coefficients nuls N_0 et donc également non-nuls $N_{\bar{0}}$ en fonction du facteur de quantification q .

$$N_{\bar{0}}(q) = N_T - N_0(q)$$

$$N_0(q) = \sum_{i=1}^{N_H} N'_{0,i}(q)$$

où le nombre de coefficients nuls N'_0 de chacune des N_H positions (u, v) est calculé à partir des histogrammes h et du vecteur de seuils r .

$$N'_0(q) = \sum_{j=1}^{r_q} h_j \quad (48)$$

Reprenons le cas des coefficients de luminance pour la position $(u, v) = (0, 1)$ pour illustrer le calcul de N_0 . Pour $q = 3$, nous obtenons $r_{q=3} = 92$ et donc $N'_0(q = 3) = 6630$ (Fig. 37). Pour $q = 100$, nous avons $r_{q=100} = 1$ et donc $N'_0(q = 100) = 1529$.

La variable linéaire l correspond au pourcentage de coefficients non-nuls. En reprenant l'Equ. (40), l'inverse du *mapping* $\Phi_2^{-1}(q)$ devient finalement, en omettant l'instant k :

$$\Phi_2^{-1}(q) = l = \frac{N_{\bar{0}}(q)}{N_T} \quad (49)$$

Le *mapping* Φ_2 (Fig. 38) se présente sous la forme d'une table de correspondance entre l et q . Par exemple pour une image donnée (Fig. 31b), le *mapping* $q = \Phi_2(l)$ et son inverse $l = \Phi_2^{-1}(q)$ est représenté numériquement par le Tableau 3.

4.5.4 Validation de l'algorithme direct

L'algorithme direct permet de trouver le *mapping* numérique Φ_2 entre le facteur de quantification q et la variable linéaire l . Cet algorithme est validé avec l'estimation de la taille de l'image compressée

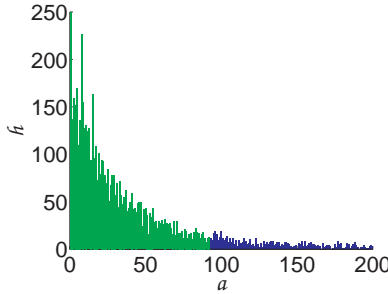


Fig. 37: Somme de toutes les valeurs qui auraient été égales à zéro après quantification pour un facteur $q = 3$ et donc $r_{q=3} = 92$. La somme jusqu'à 92 (en évidence en vert) donne $N'(q = 3) = 6630$.

q	0	1	2	3	4	5	6	7	8	9	10	11	12
l	0.005	0.005	0.010	0.012	0.013	0.014	0.015	0.016	0.018	0.019	0.020	0.021	0.022
q	13	14	15	16	17	18	19	20	21	22	23	24	25
l	0.023	0.023	0.024	0.025	0.026	0.027	0.028	0.029	0.029	0.030	0.031	0.032	0.032
q	26	27	28	29	30	31	32	33	34	35	36	37	38
l	0.033	0.034	0.035	0.035	0.036	0.037	0.038	0.038	0.039	0.040	0.041	0.041	0.042
q	39	40	41	42	43	44	45	46	47	48	49	50	51
l	0.042	0.043	0.044	0.044	0.045	0.046	0.046	0.047	0.048	0.048	0.049	0.050	0.050
q	52	53	54	55	56	57	58	59	60	61	62	63	64
l	0.051	0.052	0.052	0.053	0.054	0.055	0.056	0.056	0.057	0.058	0.059	0.060	0.061
q	65	66	67	68	69	70	71	72	73	74	75	76	77
l	0.062	0.064	0.065	0.066	0.067	0.069	0.071	0.072	0.074	0.077	0.077	0.079	0.082
q	78	79	80	81	82	83	84	85	86	87	88	89	90
l	0.084	0.086	0.089	0.093	0.096	0.100	0.103	0.108	0.112	0.117	0.124	0.129	0.137
q	91	92	93	94	95	96	97	98	99	100			
l	0.146	0.155	0.168	0.185	0.206	0.241	0.289	0.370	0.524	0.641			

Tableau 3: Table de correspondance entre q et l du *mapping* Φ_2 pour une image donnée.

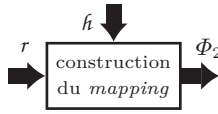


Fig. 38: Le *mapping* Φ_2 est calculé à partir des histogrammes h et des vecteurs de seuils r .

o à partir du facteur de quantification q et de la métrique de contenu c grâce à l'Equ. (27).

La taille de l'image compressée est estimée à partir de l'algorithme direct pour les quatre images de test (Fig. 31). La Fig. 39 montre l'estimation \hat{o} et la mesure o de la taille de l'image compressée. La taille o est mesurée après avoir compressé l'image avec chaque facteur de quantification (de 0 à 100). L'estimation \hat{o} est obtenue à partir de la métrique de contenu c de l'image mesurée une fois pour toutes, et du *mapping* Φ_2 calculé en une passe. La taille estimée est proche de la taille mesurée, comme le montre l'erreur $e = \frac{|o-\hat{o}|}{o}$ présentée dans la Fig. 40. L'estimation est spécialement de bonne qualité pour des valeurs de quantification intermédiaires, par exemple entre 10 et 90. Ces mesures confirment que le modèle linéaire est applicable à ces images.

4.6 Application et résultats

La méthode du *mapping* permet de déterminer la valeur des paramètres d'encodage f et q qui correspond à l'entrée unique u liée au débit de référence. Pour ce faire, le *mapping* Φ permet de passer de la fonction réalisable h dans l'espace linéaire (l'entrée unique u correspond à la pente d'une droite) à la fonction réalisable g dans l'espace d'encodage (engendré par les paramètres f et q).

Cette section présente l'utilisation de la méthode du *mapping* pour un encodeur de type M-JPEG. La Sect. 4.6.1 illustre la possibilité

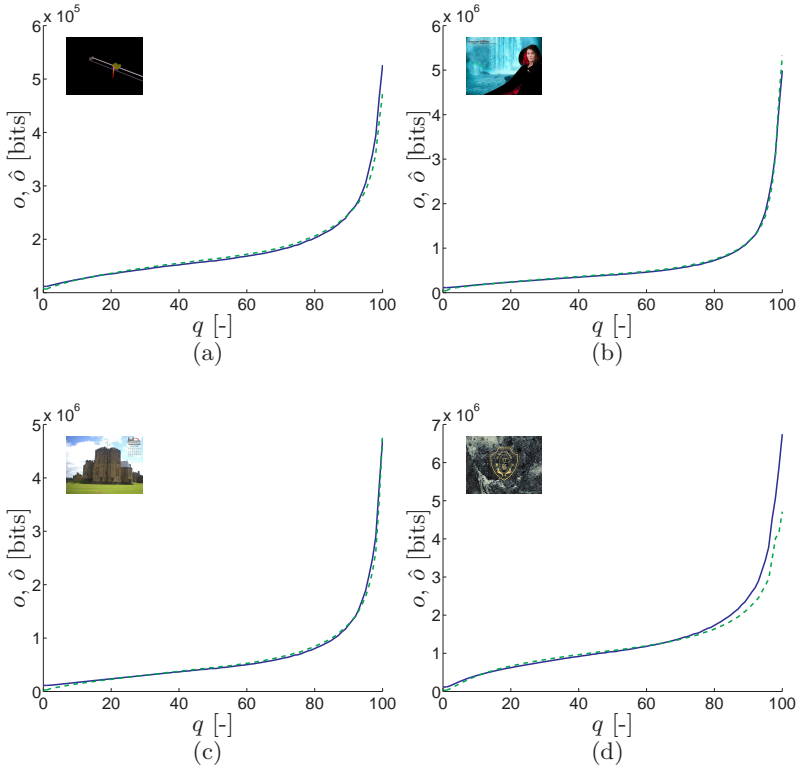


Fig. 39: Estimation (vert traitillé) et mesure (bleu continu) de la taille de l'image compressée pour les quatre images de test.

de choisir entre les combinaisons de paramètres d'encodage pour obtenir un débit constant. La Sect. 4.6.2 présente la combinaison entre la méthode du *mapping* et le contrôle de débit (Chap. 3), avec

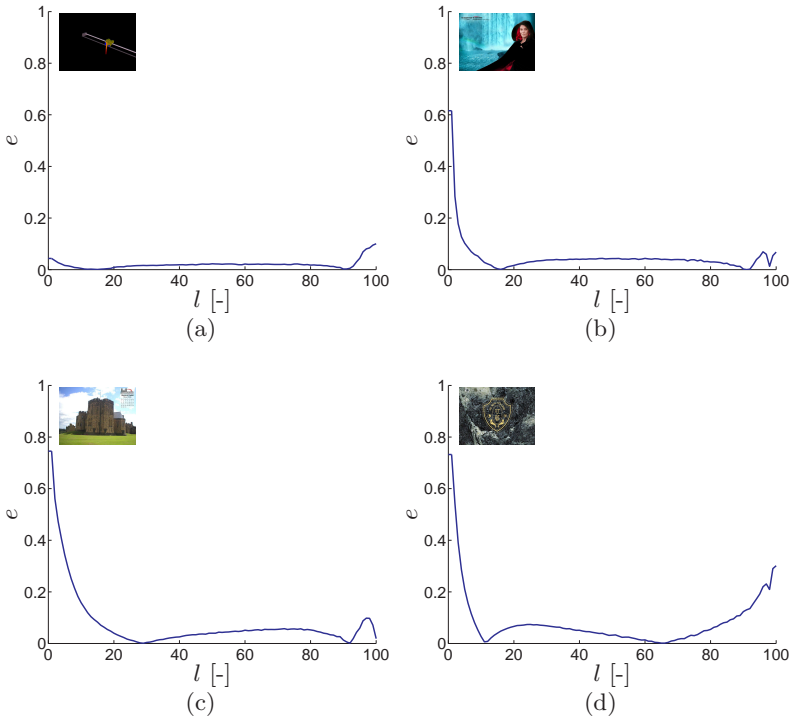


Fig. 40: Erreur d'estimation de l'image compressée pour les quatre images de test : $e = \frac{|o-\hat{o}|}{o}$.

la conception du contrôleur de type intégrateur et les résultats de mesures expérimentales sur différents flux vidéos.

4.6.1 Choix des paramètres d'encodage

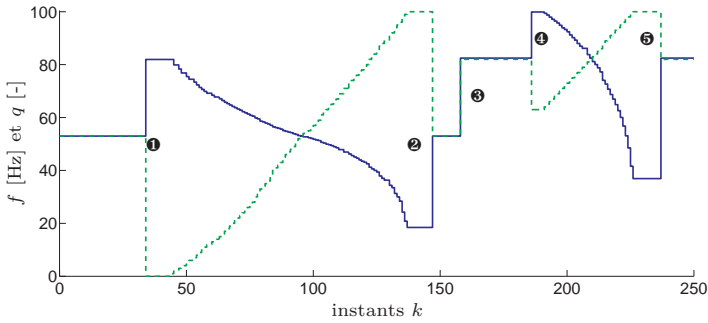
Pour une valeur de l'entrée unique donnée u liée au débit de référence, il y a plusieurs combinaisons de paramètres d'encodage possibles. Les paramètres d'encodage doivent être choisis sur la fonction réalisable g paramétrée par l'entrée unique u . Les Fig. 41 et 42 présentent le choix des paramètres d'encodage pour un flux vidéo statique (contenu constant de la Fig. 31a).

La Fig. 41 montre le comportement du débit et des paramètres d'encodage lorsque ces derniers évoluent arbitrairement sur la fonction réalisable g . Du début jusqu'à l'instant ③, l'entrée unique vaut $u = 1.5$; ensuite sa valeur change $u = 3$.

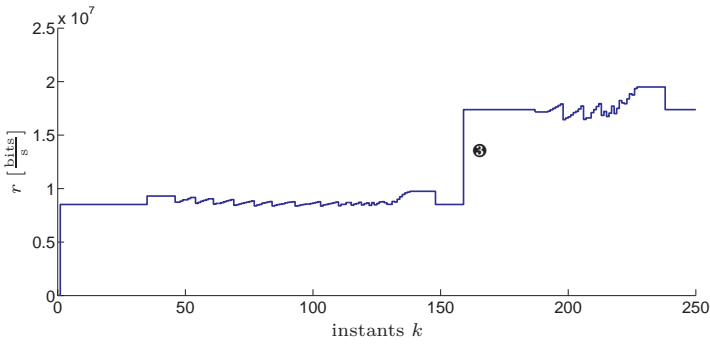
Du début jusqu'à l'instant ①, les paramètres d'encodage f et q sont choisis sur la courbe $g = 1.5$ de telle manière que leurs valeurs soient également réparties. De l'instant ① à l'instant ②, les paramètres d'encodage sont choisis le long de la courbe $g = 1.5$ en partant de la valeur de fréquence la plus grande possible jusqu'à la valeur de fréquence la plus faible possible (Fig. 42).

De l'instant ② à l'instant ④, les paramètres d'encodage sont à nouveau également répartis. A l'instant ③ la valeur de l'entrée unique a augmenté à $u = 3$; les paramètres doivent être choisis sur la fonction $g = 3$ (Fig. 42). Cela a pour effet d'augmenter les valeurs des paramètres d'encodage. De l'instant ④ à l'instant ⑤, les valeurs des paramètres d'encodage sont choisis le long de la nouvelle courbe $g = 3$ pour une fréquence f décroissante (Fig. 42).

La Fig. 41b montre que le débit résultant du choix des paramètres d'encodage f et q reste quasiment constant. Les variations du débit sont dues aux non-linéarités du système encodeur modélisé linéaire, pour la relation entre la variable l et la taille de l'image compressée o . Le contrôleur permettra d'éliminer ces variations ainsi que de forcer le débit à suivre un débit de référence désiré.



(a) Valeur des paramètres d'encodage : la fréquence f (bleu continu) et le facteur de quantification q (vert traitillé).



(b) Mesure du débit du flux vidéo généré.

Fig. 41: Evolution temporelle du débit et des paramètres d'encodage lorsque la valeur de l'entrée unique u change.

4.6.2 Contrôle de débit

Le modèle de l'encodeur pour le contrôle de débit présenté au Chap. 3 permet de suivre un débit de référence en dépit de la variation du contenu et des non-linéarités du système encodeur. Cette section présente la conception du contrôleur ainsi que des simulations en boucle fermée sur des flux vidéo statique et dynamique.

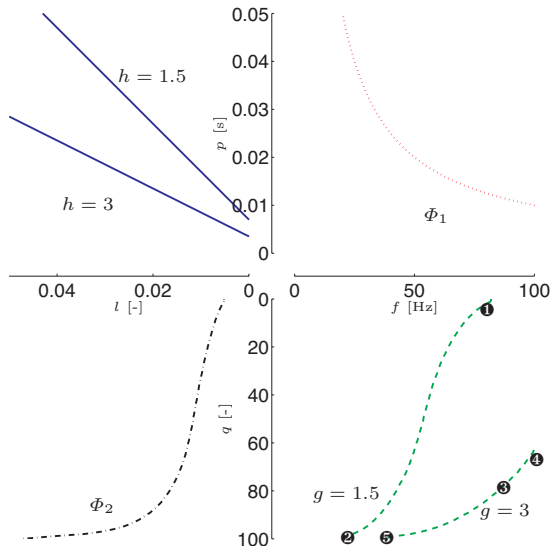


Fig. 42: La fonction réalisable $h = u$ (bleu continu) dans l'espace linéaire correspond à une droite dont la pente vaut u . La fonction réalisable $g = u$ (vert traitillé) dans l'espace d'encodage correspond au *mapping* Φ de la fonction $h = u$, pour deux valeurs de l'entrée unique $u = 1.5$ et $u = 3$.

Conception du contrôleur

Différentes stratégie de contrôle linéaire peuvent être mises en place pour contrôler le système encodeur modélisé de manière abstraite, selon l'Equ (11). Le contrôleur est synthétisé utilisant l'hypothèse courante [48, 59, 60] que le contenu $c(k)$ est constant et de valeur inconnue.

Nous utilisons un contrôleur de type intégrateur. La valeur de son gain a doit être choisie de telle sorte que le système en boucle fermée soit stable entrée bornée sortie bornée (BIBO pour *Bounded Input Bounded Output*), en fonction des valeurs de la métrique de contenu c , et plus précisément de sa valeur maximale c_{max} . Afin de garantir la stabilité, le gain est borné par le gain limite a_{lim} : $0 < a < a_{lim} = \frac{2}{c_{max}}$. Le gain choisi détermine les performances (par exemple le temps d'établissement) du système.

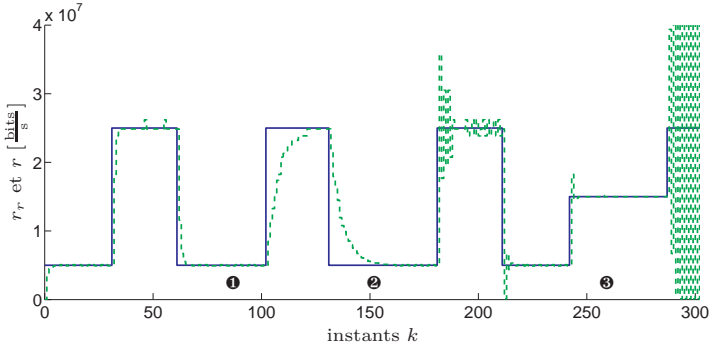
Flux vidéo statique

Afin de présenter les notions liées au *mapping*, le contrôle de débit est tout d'abord appliquée à un flux vidéo dont l'image reste statique, avec donc une métrique de contenu c fixe (Fig. 31a).

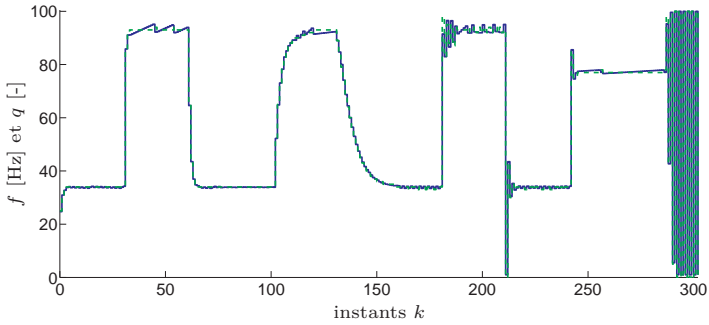
La Fig. 43 montre le comportement du débit contrôlé avec un régulateur de type intégrateur, ceci pour différentes valeurs du gain a . Le débit de référence r_r varie en forme de signal carré. Les paramètres d'encodage sont choisis sur la fonction réalisable g de façon à ce que leurs valeurs soient également réparties. Comme présenté dans la section précédente, n'importe quel autre choix de paramètres d'encodage sur la fonction réalisable g donne un débit résultant équivalent.

La valeur du gain est modifiée au cours du temps. Pour le flux vidéo statique, la valeur de la métrique de contenu vaut environ $c \approx 10^7$. La valeur du gain limite vaut donc $a_{lim} = 2 \cdot 10^{-7}$.

Du début jusqu'à l'instant ❶, le gain est choisi à $a = 0.4a_{lim}$; la réponse est rapide et ne présente pas de dépassement. De l'instant ❶ à l'instant ❷, la valeur du gain est diminuée à $a = 0.1a_{lim}$; comme



(a) Valeur du débit de référence (bleu continu) et mesure du débit (vert traitillé).



(b) Valeur des paramètres d'encodage : la fréquence f (bleu continu) et le facteur de quantification q (vert traitillé).

Fig. 43: Evolution temporelle du débit et des paramètres d'encodage en boucle fermée.

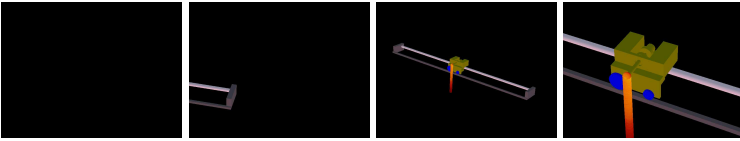
nous pouvions nous y attendre, la réponse est bien plus lente. De l'instant ② à l'instant ③, la valeur du gain est augmentée à $a = 0.8a_{lim}$; la réponse présente de grands dépassements de consigne. Dès l'instant ③, le gain est augmenté au delà du gain limite $a = 1.1a_{lim}$; le système devient instable dès qu'il y a un changement de référence.

Flux vidéo dynamique

Lorsque le contenu du flux vidéo évolue au cours du temps, le *mapping* Φ_2 doit être recalculé pour chaque image. Cette section présente la méthode du *mapping* appliquée à un flux vidéo de référence (Fig. 44a). La Fig. 44b montre l'évolution du débit pour des valeurs fixes de paramètres d'encodage. La variation du débit étant très grande et la bande passante limitée, il est nécessaire d'utiliser une mécanique pour contrôler le débit afin d'éviter des délais ou des pertes de données.

Dans le flux vidéo pris en considération, la métrique de contenu varie entre $c_{min} = 7.6 \cdot 10^6$ et $c_{max} = 1.1 \cdot 10^7$. Le contrôle de débit est appliqué à ce flux, avec une valeur de gain intégral $a = 0.4a_{lim}$ qui permet d'atteindre le débit de référence rapidement et sans dépassement. La Fig. 45 montre son efficacité pour un débit de référence variable.

Le choix des paramètres d'encodage évolue au cours du temps. Du début jusqu'à l'instant ①, les paramètres d'encodage sont choisis de valeurs également réparties. A partir de l'instant ①, la valeur des paramètres d'encodage évoluent sur la fonction réalisable g . De l'instant ① à l'instant ②, la fréquence est augmentée et la quantification réduite jusqu'à obtenir la fréquence maximale. De l'instant ② jusqu'à la fin, la quantification est augmentée et la fréquence diminuée jusqu'à obtenir la quantification maximale.



(a) Images du flux vidéo de référence aux instants ①, ②, ③ et ④.

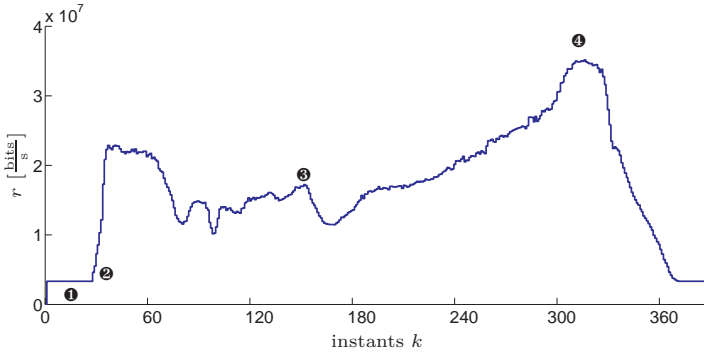
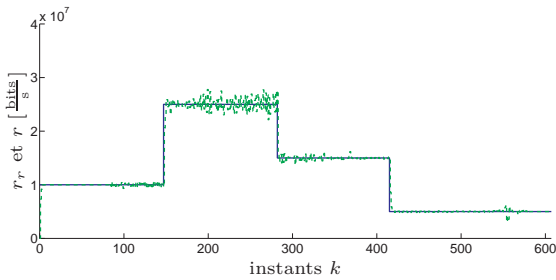
(b) Variation du débit pour des valeurs fixes de paramètres d'encodage ($f = 30$, $q = 100$) pour le flux vidéo de référence échantillonné régulièrement.

Fig. 44: Le débit du flux vidéo varie fortement lorsque les paramètres d'encodage sont constants.

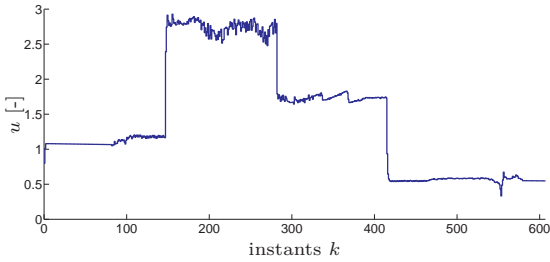
Discussion

La méthode du *mapping* a été appliquée à un flux vidéo statique puis dynamique. Grâce à cette méthode, le contrôle de débit permet de suivre un débit de référence de manière efficace. Les paramètres d'encodage peuvent être choisis librement sur la fonction réalisable g .

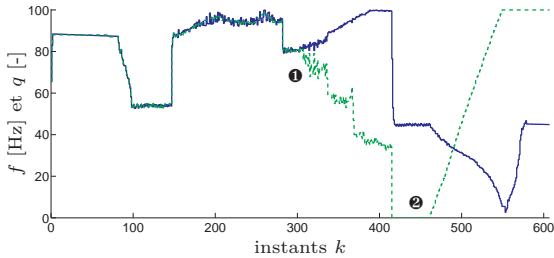
Le contrôleur de type intégral permet de contrôler le modèle abstrait sous l'hypothèse d'une métrique de contenu constante. Cette hypothèse peut être relaxée en pratique car le contrôleur choisi intègre l'erreur. En utilisant l'encodeur JPEG pour compresser les



(a) Valeur du débit de référence (bleu continu) et mesure du débit (vert traitillé).



(b) Entrée unique u calculée par le contrôle de débit.



(c) Valeur des paramètres d'encodage : la fréquence f (bleu continu) et le facteur de quantification q (vert traitillé).

Fig. 45: Comportement en boucle fermée pour le flux vidéo dynamique échantillonné différemment. Les paramètres d'encodage f et q sont choisis tels qu'ils sont sur la fonction g correspondant à l'entrée unique u .

images, la métrique de contenu c varie faiblement. Mais avec d'autres encodeurs de type DCT, la métrique de contenu peut varier fortement [59]. Le contrôleur permet de suivre la référence avec un contenu inconnu et de compenser les non-linéarités du système.

Le contrôle adaptatif permettrait d'améliorer les performances en adaptant la valeur du gain de l'intégrateur $a(k)$ en fonction de la mesure du contenu $c(k)$.

4.7 Conclusion

Dans ce chapitre, nous avons présenté une méthode qui permet de passer de l'entrée unique u , issue du contrôle de débit (Chap. 3), aux paramètres d'encodage propres à l'encodeur réel, la fréquence f et la quantification q . La fonction réalisable g couvre les combinaisons des paramètres d'encodage qui correspondent à l'entrée unique u . Le *mapping* Φ permet de trouver g dans l'espace d'encodage à partir d'une fonction connue h . Le contrôle de débit conçu pour un modèle abstrait de l'encodeur peut ainsi être appliqué concrètement.

L'analyse dans le ρ -domaine permet de faire correspondre le paramètre de quantification q à une variable l proportionnelle à la taille de l'image compressée. Nous avons proposé le *mapping* numérique Φ_2 pour déterminer cette correspondance. Comme ce *mapping* dépend du contenu de l'image, il doit être évalué à chaque itération de l'encodeur pour éviter des délais supplémentaires. Pour ce faire, nous avons développé un nouvel algorithme rapide à une passe basé sur les caractéristiques de l'encodeur.

La méthode du *mapping* permet de prendre en compte plusieurs paramètres d'encodage simultanément. La flexibilité donnée par un degré de liberté supplémentaire permet d'obtenir une plus large échelle de compression afin d'atteindre par exemple des débits très faibles. L'implémentation temps réel de cette flexibilité et/ou d'un contrôleur linéaire pour le débit est possible grâce à l'algorithme direct proposé.

Notons que cette méthode du *mapping* est générique car elle permet de transposer l'entrée unique u dans l'espace d'encodage, indépendamment des paramètres d'encodage choisis. Son utilisation nécessite tout d'abord de représenter la fonction réalisable h dans un espace naturel pour l'entrée unique u ; dans notre cas, il s'agit de l'espace linéaire où u représente la pente de h . Ensuite, il faut déterminer le *mapping* Φ pour trouver la fonction réalisable g dans l'espace d'encodage, dont la dimension dépend du nombre de paramètres d'encodage sélectionnés. Il n'y a pas de méthode générale pour déterminer ce *mapping* Φ pour tout type d'encodeur. Ce chapitre a présenté la méthode pour les encodeurs basés sur la DCT, comme c'est le cas couramment.

La méthode du *mapping* a été combinée au contrôle de débit puis appliquée sur un flux vidéo encodé en M-JPEG. Les mesures confirment que l'algorithme permet de suivre efficacement une consigne de débit, malgré des changements de contenu du flux vidéo.

Le *mapping* développé dans ce chapitre est utilisé par le schéma d'adaptation présenté dans le Chap. 5. Grâce à l'intégration du modèle de QoE présenté dans le Chap. 2, les paramètres d'encodage sont adaptés pour obtenir la meilleure expérience utilisateur possible.

Adaptation

Les paramètres d'encodage sont adaptés en temps réel pour maximiser la qualité d'expérience, grâce au modèle et à la mesure des variables contextuelles.

Rappelons que l'objectif de cette thèse est de déterminer comment adapter les paramètres d'encodage d'un flux vidéo de AR, pour obtenir la meilleure expérience utilisateur possible en fonction d'un contexte dynamique. Dans le Chap. 2, nous avons présenté le modèle de QoE qui représente l'impact des paramètres d'encodage (fréquence et quantification) et des variables contextuelles (délai et vitesse de la tête) sur la métrique QoE. Nous avons montré que pour un contexte donné, l'expérience utilisateur change en fonction de la combinaison des paramètres d'encodage choisis. Dans le Chap. 3, nous avons présenté l'algorithme de contrôle de débit qui détermine la valeur de l'entrée unique du modèle de l'encodeur. Dans le Chap. 4, nous avons présenté la méthode du *mapping* (mise en correspondance) qui permet de déterminer la fonction réalisable dans l'espace d'encodage. Cette fonction représente les combinaisons des paramètres d'encodage qui correspondent à la valeur de l'entrée unique issue du contrôle de débit.

Dans ce chapitre, nous proposons le schéma d'adaptation global qui détermine en temps réel la combinaison optimale des paramètres d'encodage afin de maximiser la QoE. Le modèle de QoE et la fonction réalisable étant représentés dans le même espace d'encodage,

il est possible de les combiner pour trouver les valeurs optimales des paramètres d'encodage pour le contexte et le débit de référence donnés.

La Sect. 5.1 présente l'état de l'art dans le domaine de l'adaptation de flux vidéos. L'architecture utilisée pour notre approche d'adaptation est présentée dans la Sect. 5.2. La Sect. 5.3 présente l'adaptation contextuelle qui permet de déterminer la combinaison de paramètres d'encodage qui donne la meilleure QoE possible en fonction du contexte. Notre schéma d'adaptation global est présenté et illustré dans la Sect. 5.4. Les éléments des chapitres précédents y sont combinés à l'adaptation contextuelle. La Sect. 5.5 présente l'application du schéma d'adaptation en simulation. Des mesures expérimentales permettent de valider le schéma d'adaptation proposé. Finalement, la Sect. 5.6 résume les éléments principaux de ce chapitre.

5.1 Etat de l'art

Des stratégies d'adaptation sont utilisées pour garantir ou maximiser la qualité d'un flux vidéo en fonction du débit. En fonction des besoins, l'adaptation se fait au niveau de l'encodage et/ou de la transmission.

De nombreuses méthodes d'adaptation ont été développées afin de contrôler la qualité d'un flux vidéo. Dans le but de contrôler la qualité vidéo, un modèle du système visuel humain (HVS pour *Human Visual System*) est utilisé pour prédire la distorsion de la vidéo due à la transmission [61]. Pour pouvoir être utilisées en rétroaction, des métriques subjectives ont été spécialement développées pour l'adaptation [62, 63].

Les transmissions sur un réseau sans fil sont soumises à de larges variations de caractéristiques du réseau. Certaines techniques d'adaptation pour la transmission sont conçues pour garantir une qualité de service (QoS pour *Quality of Service*). Le transcodage du

flux vidéo, la modification de la fréquence et de la profondeur de couleurs permettent de réduire les pertes de données et ainsi de maintenir la QoS requise [64]. La planification d'envoi de paquets permet de garantir un faible taux de perte et ainsi d'augmenter la qualité de la vidéo transmise [65]. Il est également possible de planifier la transmission au niveau de l'application afin d'adapter le débit de la vidéo en fonction de l'estimation de la bande passante disponible et de respecter des contraintes de délai [66,67]. Ces études permettent d'améliorer les performances de la transmission, cependant la notion d'expérience utilisateur n'est pas prise en compte.

Cranley et al. ont développé un schéma d'adaptation pour le *streaming* (diffusion de flux continu) de vidéo en tenant compte de la qualité perçue par l'utilisateur, l'encodage vidéo et la transmission. Le délai engendré par le tampon du mécanisme de *streaming* n'est pas acceptable pour notre domaine d'application d'AR. La contribution principale de Cranley concerne la modélisation de la qualité au travers de la trajectoire d'adaptation optimale (OAT pour *Optimal Adaptation Trajectory*), qui permet de lier les paramètres d'encodage à la qualité (Sect. 2.1), avec de nombreuses évaluations subjectives. L'OAT proposée par Cranley ne considère que deux paramètres d'encodage : la fréquence et la résolution spatiale. Dans leurs perspectives, ils proposent d'ailleurs de traiter plus de paramètres ; ce qui est le cas de notre modèle de QoE qui considère les variables contextuelles. Une OAT est proposée pour chaque vidéo ayant une dynamique différente. Aucune solution générique n'est proposée pour gérer la dynamique d'une nouvelle vidéo. Dans notre cas spécifique à l'AR, la dynamique est quantifiée avec la vitesse de la tête. Cela permet de gérer n'importe quelle nouvelle scène de manière générique. De plus, la structure du modèle de QoE permet d'intégrer facilement d'autres métriques de dynamique. Cranley et al. proposent l'algorithme adaptation perceptuelle de qualité (PQA pour *Perceptual Quality Adaptation*) pour indiquer comment l'encodage doit être adapté en fonction de l'OAT [68]. Leur stratégie d'adaptation est relativement basique, car elle ne fait que changer de source vidéo pré-encodée en fonction de la bande passante. Les

simulations montrent que l'utilisation explicite de la qualité pour les décisions d'adaptation permettent de maximiser la qualité perçue par l'utilisateur [69]. En ne traitant pas les paramètres de compression (par exemple la quantification JPEG (Joint Photographic Experts Group)), Cranley et al. occultent le problème du contrôle de débit. Leur stratégie d'adaptation n'est pas capable de suivre un débit de référence, mais peut uniquement contrôler grossièrement le débit dans une fourchette de valeurs. Le contrôle de débit combiné à la méthode de *mapping* que nous avons proposée permet de générer efficacement des paramètres d'encodage propres à un encodeur. Cranley et al. utilisent des mécanismes et des protocoles spécifiques. Notre approche système permet de se détacher des mécanismes implantés et d'intégrer de nouveaux éléments de manière générique.

5.2 Architecture

Afin de maximiser la QoE en fonction du contexte dynamique, nous proposons une approche globale d'adaptation, allant de la création du flux vidéo jusqu'à sa perception par l'utilisateur (Fig. 46).

Les éléments de cette infrastructure sont représentés par une architecture client-serveur qui intègre également l'utilisateur (Fig. 46a). Le client mobile comporte l'interface de rendu pour l'utilisateur, alors que le serveur gère l'adaptation et les processus nécessitant une grande puissance de calcul [70]. Il s'agit d'une stratégie dite à client léger¹ qui permet de réduire l'équipement mobile au strict nécessaire. Le flux vidéo (traits larges) est transmis du serveur jusque vers l'utilisateur. Certains signaux (traits fins) sont nécessaires pour la génération de la scène d'AR et pour l'adaptation (Fig. 46b) :

$m_c(k)$: Les mesures contextuelles correspondent aux valeurs mesurables du contexte, soit la *pose* (combinaison de position et orientation) de la tête et le délai global d . Le flux vidéo est

1. *thin client*

généralisé à partir de la scène d'AR et du point de vue de l'utilisateur donné par la *pose*. La vitesse de la tête s vient de la discrétisation de l'orientation de la tête, basé sur les quaternions. Le délai $d(k)$ et la vitesse $s(k)$ sont utilisés à chaque itération k de l'adaptation.

UX : L'expérience utilisateur correspond à l'appréciation subjective donnée par l'utilisateur. Elle est utilisée pour identifier le modèle de QoE mais n'est plus requise en temps réel pour l'adaptation.

$n(k)$: Les métriques de la QoS de la transmission sont généralement combinées à partir des informations obtenues par le serveur et le client.

$r(k)$: La mesure du débit provient du système encodeur qui est situé au niveau du serveur. Le débit est mesuré à chaque instant discret k .

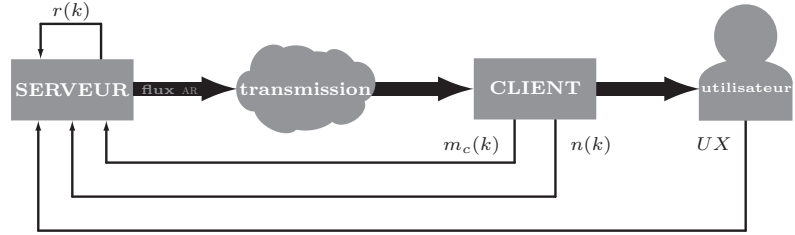
La représentation système permet de dissocier les signaux et les flux. Ainsi, l'adaptation contrôle le système AR en adaptant les paramètres d'encodage $v_e(k)$ (Fig. 46c). Le système AR comprend l'infrastructure globale, de l'imagerie à l'utilisateur, en faisant abstraction de l'architecture.

La section suivante présente le schéma d'adaptation global en détail. Le schéma global intègre les éléments présentés dans les chapitres précédents.

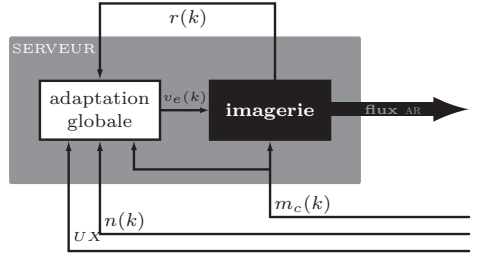
5.3 Adaptation contextuelle

L'adaptation est composée de trois boucles de rétroaction et peut être représenté sous forme compacte avec le système AR (Fig. 47).

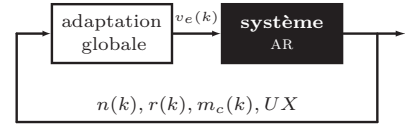
- L'adaptation de la transmission (boucle externe inférieure) : Les métriques de la QoS de la transmission sont utilisées par le contrôleur de la transmission pour obtenir un débit de référence $r_r(k)$ (Sect. 3.2.2). Cet élément n'est pas traité dans le cadre de cette thèse.



(a) Le serveur utilise différentes mesures pour générer un flux vidéo adapté au contexte.



(b) Le serveur est constitué des processus d'adaptation et d'imagerie. L'imagerie génère, encode et envoie le flux vidéo.



(c) Représentation compacte de l'adaptation. Le système AR englobe l'imagerie, la transmission, le client et l'utilisateur.

Fig. 46: L'approche d'adaptation proposée utilise une architecture client-serveur. Le flux vidéo d'AR est généré et encodé par le serveur, puis transmis au client et affiché pour l'utilisateur. Les blocs gris représentent des entités physiques, les blocs noirs représentent des systèmes et les blocs blancs représentent des mécanismes d'adaptation.

- Le contrôle de débit (boucle interne inférieure) détermine la valeur de l'entrée unique $u(k)$ à partir du débit mesuré $r(k)$ et du débit de référence (Chap. 3). La méthode du *mapping* donne les combinaisons de paramètres d'encodage possibles, représentés dans la fonction réalisable $g(v_c, k)$ (Chap. 4).
- L'adaptation contextuelle (boucle supérieure) permet de choisir la combinaison de paramètres d'encodage afin de maximiser la QoE de l'utilisateur en fonction du contexte.

Cette section présente l'adaptation contextuelle, avec notamment l'interaction entre la fonction réalisable g et le modèle de QoE (Chap. 2).

5.3.1 Fonctions réalisables

Les fonctions réalisables g (Déf. 7) et h (Déf. 10) jouent un rôle prépondérant dans le schéma d'adaptation. Elles permettent de faire le lien entre le contrôle de débit et le modèle de QoE.

La fonction réalisable h est définie dans l'espace linéaire (Déf. 9), alors que la fonction réalisable g est définie dans l'espace d'encodage (Déf. 3). Pour rappel, les paramètres d'encodage (la fréquence f et la quantification q) doivent être choisis sur la courbe définie par la fonction g .

Le schéma à quatre quadrants permet de représenter ces fonctions ainsi que le *mapping* Φ qui les lie (Fig. 48a). La fonction réalisable g dans l'espace d'encodage évolue au cours du temps lorsque l'entrée unique et/ou le contenu varient.

5.3.2 Espace d'encodage augmenté

L'espace d'encodage augmenté (Déf. 3) permet de lier la fonction réalisable au modèle de QoE. Cet espace correspond à l'espace engendré par les paramètres d'encodage (fréquence et quantification) et la métrique QoE. Le modèle de QoE contextuel M_c (modèle pour des valeurs définies des variables contextuelles) est défini dans cet espace.

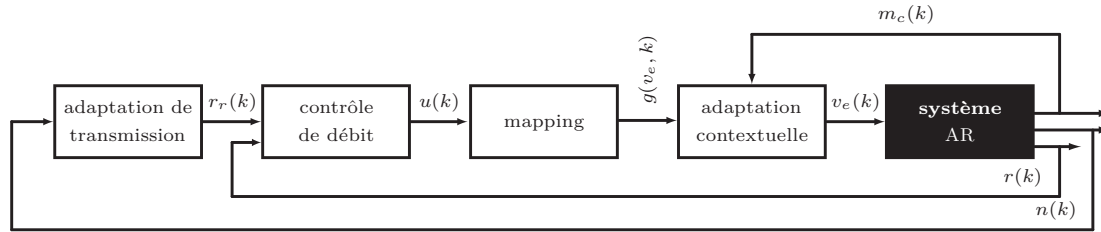


Fig. 47: L'adaptation contextuelle détermine la combinaison de paramètres d'encodage $v_e(k)$ qui maximisent la QoE. Les mesures contextuelles $m_c(k)$ et le modèle de QoE sont utilisés pour déterminer la combinaison optimale choisie sur la courbe de la fonction g .

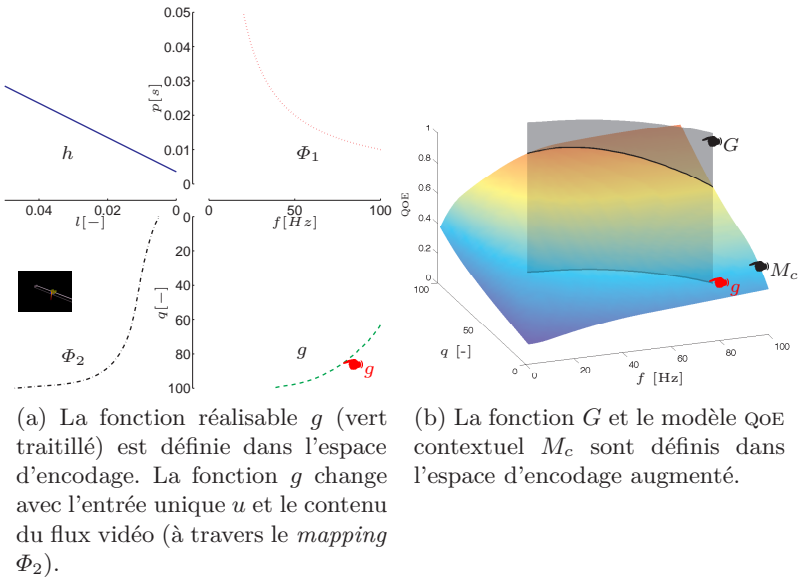


Fig. 48: La fonction réalisable augmentée G correspond à la surface verticale issue de la fonction réalisable g . Dans ces figures, l'entrée unique vaut $u = 2$.

La fonction réalisable g est quant à elle définie dans l'espace d'encodage (pas augmenté). Nous définissons la fonction réalisable augmentée G dans l'espace d'encodage augmentée comme étant la surface verticale issue de la fonction g (Fig. 48b). Ainsi, le modèle de QoE contextuel et la fonction réalisable augmentée G se trouvent dans le même espace.

Les sections suivantes présentent l'utilisation du modèle de QoE afin d'évaluer la valeur de la métrique QoE pour différentes combinaisons de paramètres d'encodage situés sur la fonction réalisable g .

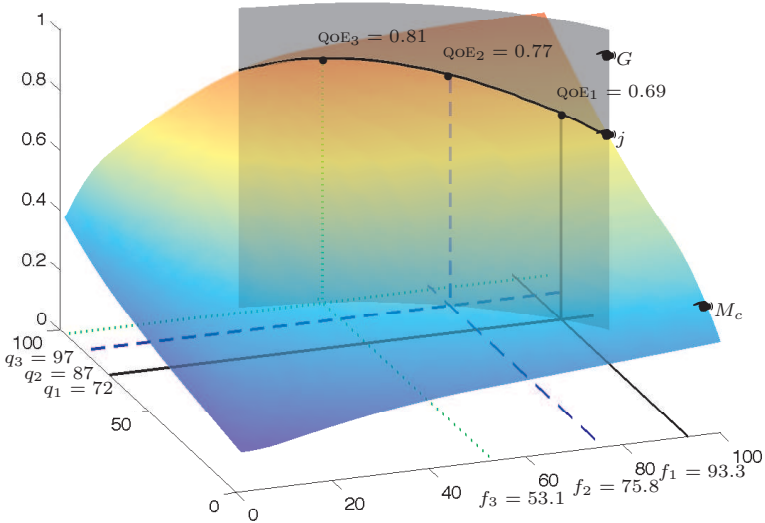


Fig. 49: Le modèle contextuel $M_c(f, q, k)$ et la fonction réalisable augmentée $G(f, q, k) = u$ sont représentés dans le même espace. Pour une combinaison de paramètre d'encodage donnée, la fonction d'intersection $j(f, q, k)$ permet d'évaluer la métrique QoE. L'évaluation de la QoE pour trois combinaisons est représentée $QoE_i = M_c(f_i, q_i, k)$.

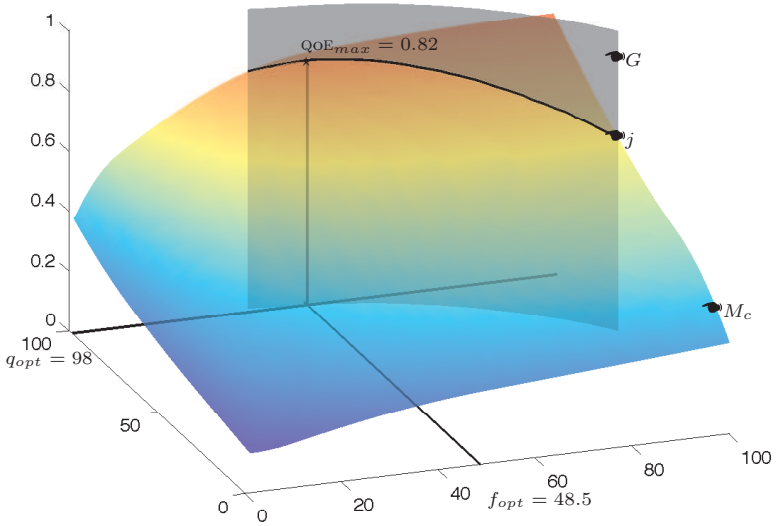


Fig. 50: L'adaptation contextuelle choisit la combinaison des paramètres d'encodage optimaux f_{opt} et q_{opt} donnant la valeur de QoE maximale.

5.3.3 Evaluation de la QoE

Les paramètres d’encodage sont choisis sur la courbe de la fonction réalisable g . Pour chaque combinaison de paramètres d’encodage, la métrique QoE correspondante peut être évaluée à partir du modèle QoE $= M(f, q, d, s)$ et des valeurs des variables contextuelles.

En fait, la métrique QoE est liée à toutes les combinaisons possibles par l’intersection entre la fonction réalisable augmentée G et le modèle contextuel M_c . Cette *intersection* est donnée par la fonction QoE $= j(f, q, k)$ (Fig. 49).

5.3.4 Optimisation

Afin de maximiser la QoE, l’adaptation contextuelle détermine la combinaison optimale des paramètres d’encodage. La combinaison optimale correspond au maximum de la fonction d’intersection j entre la fonction réalisable augmentée G et le modèle contextuel M_c (Fig. 50).

$$\operatorname{argmax}_{f,q} j(f, q, k) \quad (50)$$

5.4 Schéma d’adaptation global

Le concept d’adaptation est représenté au travers du schéma d’adaptation global (Fig. 51). Ce schéma comporte plusieurs boucles de rétroaction, articulées autour du système AR. Les prochaines sections présentent les blocs du schéma ainsi que l’algorithme en pseudo-code (Fig. 52).

5.4.1 Identification du modèle

Le modèle $M(v_e, v_c)$ est identifié au préalable à partir des mesures subjectives d’expérience utilisateur UX . (Chap. 2). Le modèle générique M_g est utilisé à la place d’un modèle identifié expérimentalement, il ne change pas durant l’expérimentation.

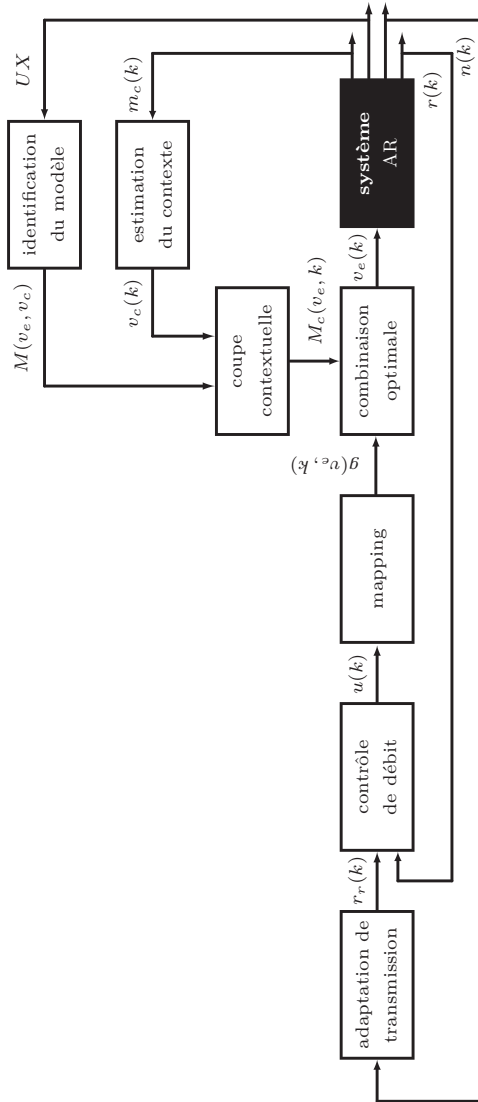


Fig. 51: Le schéma d'adaptation global maximise la QoE en temps réel.

```

% modélisation de QoE
M=identification(UX)

% boucle effectuée à chaque itération d'encodage
boucle(k) % q(k)=q(k+1)
    % mesures
    n(k)=mesure_communication()
    r(k)=mesure_débit()
    pose(k)=mesure_orientation()
    délais(k)=mesure_délais()

    % imagerie et transmission
    scène(k)=génère(pose(k))
    image(k)=encode(scène(k),q(k))
    transmission(image(k))

    % adaptation du débit
    r_r(k)=adaptation_transmission(n(k))
    u(k)=contrôle_débit_intégrateur(r_r(k),r(k))
    g(k)=mapping(u(k))

    % adaptation contextuelle
    s(k)=vitesse(pose(k))
    d(k)=somme_délais(délais(k))
    M_c(k)=coupe(M,s(k),d(k))

    % combinaison optimale
    j(k)=intersection(M_c(k),g(k))
    (f(k+1),q(k+1))=argmax(j(k))

    % attente (période définie par la fréquence)
    attendre(1/f(k+1))
fin_boucle(k)

```

Fig. 52: Pseudo-code pour l'adaptation au niveau du serveur.

5.4.2 Estimation du contexte

Les valeurs des variables contextuelles $v_c(k)$ sont estimées à partir des mesures contextuelles $m_c(k)$. L'estimation du contexte comprend l'estimation de la vitesse de la tête s à partir de l'orientation mesurée, et l'estimation du délai global d sur la base de la mesure des délais des processus internes.

5.4.3 Coupe contextuelle

Le modèle contextuel $M_c(v_e, k)$ correspond à la section du modèle $M(v_e, v_c)$ dans l'espace d'encodage augmenté (Déf. 4). Le modèle contextuel est valide pour l'instant k et ne dépend donc plus du contexte.

5.4.4 Adaptation de transmission

Sur la base des mesures du canal de communications $n(k)$, comme les métriques QoS, l'adaptation de transmission détermine le débit de référence $r_r(k)$. Le mécanisme d'adaptation de transmission n'est pas traité dans cette thèse.

5.4.5 Contrôle de débit

Grâce au modèle abstrait de l'encodeur, un contrôleur est utilisé pour contrôler le débit du flux vidéo d'AR (Chap. 3). La valeur de l'entrée unique $u(k)$ est calculée à partir du débit de référence $r_r(k)$ et du débit $r(k)$ mesuré à la sortie de l'encodeur.

5.4.6 Mapping

La méthode du *mapping* présentée au Chap. 4 permet de déterminer les combinaisons de paramètres d'encodage qui correspondent à la valeur de l'entrée unique $u(k)$. Les combinaisons possibles appartiennent à la fonction réalisable $g(v_e, k) = u(k)$. La courbe g varie en fonction du contenu du flux vidéo et de la valeur de l'entrée unique u .

5.4.7 Combinaison optimale

La combinaison de paramètres d'encodage $v_e(k+1)$ est choisie telle qu'elle maximise la métrique QoE (Sect. 5.3).

5.4.8 Système AR

Les paramètres d'encodage optimaux $v_e(k+1)$ sont appliqués au système AR par l'intermédiaire de l'encodeur. A chaque instant k , l'adaptation calcule les valeurs optimales des paramètres d'encodage. Les paramètres d'encodage pour la prochaine itération valent pour la fréquence $f(k+1)$ et pour la quantification $q(k+1)$. La prochaine image sera donc encodée avec un facteur de quantification de $q(k+1)$ dans une période de $\frac{1}{f(k+1)}$.

5.5 Application

Cette section présente l'application du schéma d'adaptation. L'effet de l'adaptation contextuelle est illustré en simulation dans la Sect. 5.5.1. Puis l'adaptation est appliquée en temps réel (Sect. 5.5.2). La performance de l'approche proposée est comparée avec d'autres scénarios où certains éléments d'adaptation sont inactifs.

5.5.1 Simulation de l'adaptation contextuelle

L'adaptation est appliquée à deux scénarios. Dans le premier, le contexte varie (vitesse de la tête et délai) et le contenu du flux vidéo est choisi constant. Dans le deuxième, le contexte reste constant mais le contenu du flux vidéo varie.

Le contexte varie (G constant, M_c varie)

Les Fig. 53 et 54 montrent comment l'adaptation contextuelle modifie les paramètres d'encodage en fonction du contexte. La Fig. 53 montre l'évolution temporelle des valeurs mesurées et des paramètres d'encodage optimaux. Afin de mieux suivre la démarche, la Fig. 54 présente le modèle contextuel à des instants donnés. Les profils des variables contextuelles sont choisis arbitrairement (Fig. 53a et 53b). La fonction réalisable g dans l'espace d'encodage est choisie constante; ce serait le cas pour un flux vidéo statique et une entrée unique u constante.

Durant les périodes ① à ③, le délai est nul (Fig. 53b). Durant la période ①, la tête est statique (Fig. 53a). Conformément à l'espace d'encodage augmenté à l'instant ① (Fig. 54a), l'adaptation contextuelle choisit un facteur de quantification maximum et une fréquence relativement faible (Fig. 53c). La QoE est à sa valeur maximale (Fig. 53d).

Durant la période ②, la vitesse de la tête augmente jusqu'à sa valeur maximale. L'adaptation contextuelle diminue la valeur de la quantification et augmente la fréquence (Fig. 53c). Les Fig. 54b et 54c montrent le modèle contextuel pour les instants ② et ③ où la vitesse de la tête est respectivement moyenne et élevée. Durant cette période, la QoE calculée à partir du modèle de QoE diminue (Fig. 53d).

Durant la période ③, la vitesse de la tête diminue jusqu'à une valeur moyenne. Ceci a pour résultat la diminution de la fréquence et l'augmentation de la quantification.

Durant la période ④, la vitesse de la tête reste constante et le délai augmente sensiblement. L'adaptation contextuelle a sélectionné la meilleure combinaison de paramètres d'encodage possible. Le modèle pour l'instant ④ est illustré sur la Fig. 54d.

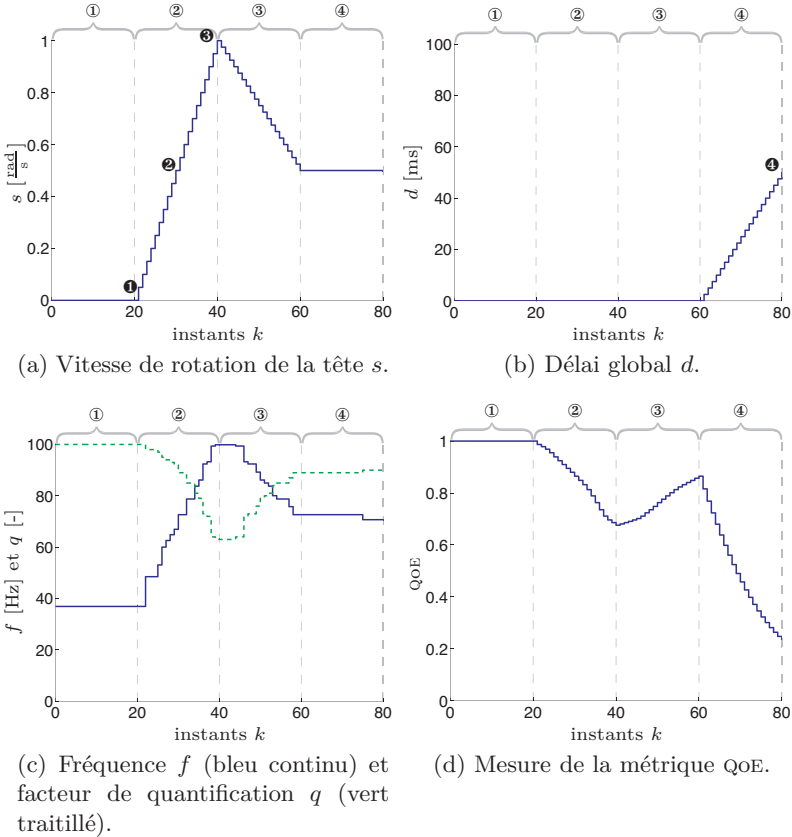
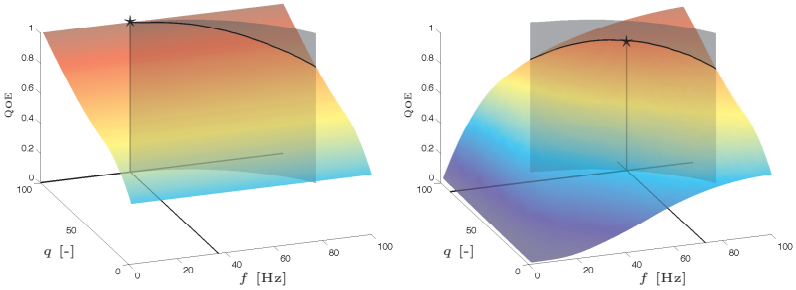
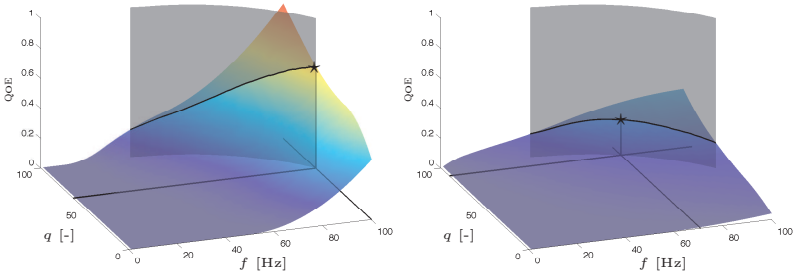


Fig. 53: L'adaptation contextuelle modifie les paramètres d'encodage, la fréquence f et la quantification q , pour obtenir la meilleure QoE possible en fonction du contexte variable. Les valeurs des variables contextuelles, la vitesse s et le délai d , sont choisies arbitrairement. La métrique QoE est obtenue à partir du modèle.



(a) A l'instant ❶, la vitesse de la tête est nulle ($s = 0$). La combinaison est optimale pour une valeur de quantification élevée.

(b) A l'instant ❷, la vitesse de la tête est moyenne ($s = 0.5$). La combinaison est optimale pour une valeur de quantification légèrement supérieure à la valeur de fréquence.



(c) A l'instant ❸, la vitesse de la tête est élevée ($s = 1$). La combinaison est optimale pour une fréquence élevée.

(d) A l'instant ❹, la vitesse de la tête est moyenne et le délai est moyen ($d = 0.5$). La combinaison optimale donne une QoE faible, mais maximale pour le contexte donné.

Fig. 54: Le modèle contextuel change lorsque les variables contextuelles évoluent. Pour ce scénario, la fonction réalisable augmentée est supposée constante. La combinaison optimale (★) des paramètres d'encodage maximise la QoE.

Le flux vidéo varie (G varie, M_c constant)

Le contenu du flux vidéo influence la fonction réalisable g . Ainsi même pour un contexte choisi constant et une entrée unique u fixe, l'adaptation modifie les paramètres d'encodage en fonction du contenu variable. La Fig. 55a montre le contenu du flux vidéo à quatre instants donnés. De l'instant ❶ à l'instant ❷, l'objet virtuel prend de plus en plus de place dans le champ de vision de l'utilisateur. De l'instant ❷ à l'instant ❸, l'objet sort progressivement du champ de vision de l'utilisateur jusqu'à en sortir totalement à l'instant ❹.

La Fig. 56 montre l'effet du contenu sur la fonction réalisable augmentée G pour les instants ❶, ❷, ❸ et ❹. Pour un modèle contextuel M_c fixe, la combinaison optimale des paramètres d'encodage évolue. La Fig. 55b montre l'évolution temporelle des paramètres d'encodage optimaux.

5.5.2 Mesures expérimentales de l'adaptation

Le schéma d'adaptation global maximise la QoE en tenant compte du contexte dynamique et du flux vidéo variable. La vitesse de la tête s est calculée par différentiation des mesures réelles d'orientation de l'utilisateur. Le délai global d évolue en fonction du débit du flux r et du débit de référence r_r sur la base du modèle de la transmission (Sect. 3.2.2).

Dans la Fig. 57, nous présentons différents scénarios d'adaptation et comparons la métrique QoE résultante. Comme les scénarios utilisent des fréquences d'échantillonnage différentes, la comparaison est présentée dans le domaine temporel t , où les différentes valeurs subissent un maintien d'ordre zéro entre les échantillons discrets. Le scénario de référence comporte tous les éléments du schéma d'adaptation global. La Fig. 57a montre l'évolution de la vitesse de la tête et du délai global pour ce scénario. La vitesse de la tête varie fortement ; ceci est dû à des mouvements courts et brusques de l'utilisateur. Le délai est faible puis augmente rapidement ; ceci est dû à une chute

du débit de référence r_r à l'instant ③, que le contrôle de débit ne corrige pas instantanément. La Fig. 57b montre l'évolution de la QoE pour les quatre scénarios pris en compte :

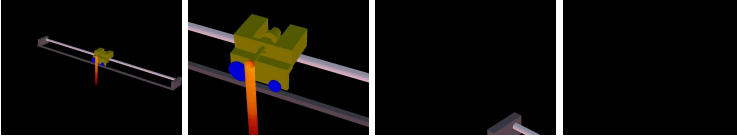
1. Schéma global avec adaptation contextuelle. Les paramètres d'encodage sont choisis de telle manière qu'il maximisent la QoE en fonction du contexte courant.
2. Schéma global avec quantification q maximale.
3. Schéma global avec fréquence f maximale.
4. Schéma global avec les paramètres d'encodage ayant la même valeur.

Le scénario de référence qui intègre l'adaptation contextuelle obtient clairement une meilleure QoE que les autres scénarios. Il arrive cependant que la QoE d'un autre scénario soit temporairement plus élevé que lors du scénario de référence. Cela est dû au fait que les scénarios ont été effectués séparément. Par exemple, l'instant discret k auquel la vitesse de la tête et le délai sont mesurés peut être différent ; ce qui influence directement la QoE.

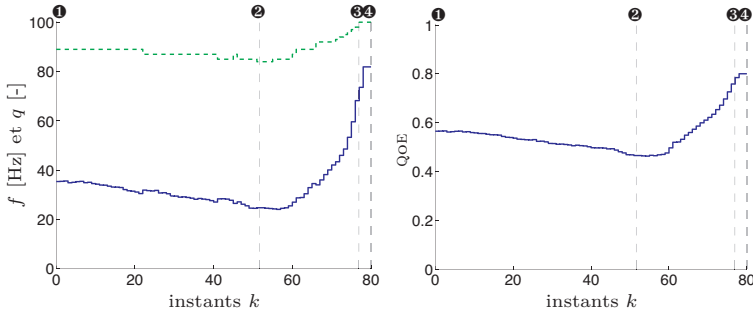
Le scénario avec une quantification q maximale obtient des résultats proches du scénario de référence, sauf quand la vitesse de la tête est élevée (par exemple à l'instant ②). Le scénario avec une fréquence f maximale donne une QoE nettement plus faible que le scénario de référence lorsque la vitesse de la tête est faible. L'écart de QoE entre ces deux scénarios est plus faible lorsque le débit est plus élevé, entre l'instant ① et l'instant ③. Le scénario avec les paramètres d'encodage ayant les mêmes valeurs obtient une QoE proche du scénario de référence, mais toujours plus basse.

Le scénario de référence avec l'adaptation contextuelle des paramètres d'encodage est donc le meilleur des quatre scénarios présentés. Cela montre la pertinence de l'approche d'adaptation proposée.

Si un autre modèle de QoE est utilisé, l'évolution des paramètres d'encodage serait différent, mais l'adaptation contextuelle permettrait toujours d'obtenir la meilleure QoE possible grâce à l'utilisation du modèle.



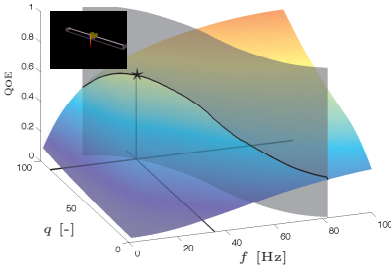
(a) Contenu du flux vidéo dynamique aux instants ①, ②, ③ et ④. Le contenu devient de plus en plus riche jusqu'à l'instant ②, puis le contenu s'appauvrit jusqu'à obtenir une image uniformément noire à l'instant ④.



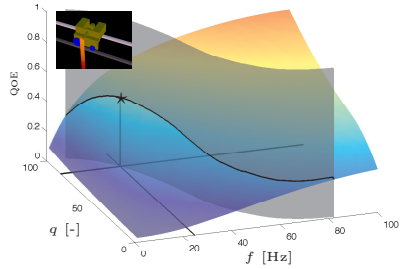
(b) Fréquence f (bleu continu) et facteur de quantification q (vert traitillé).

(c) Mesure de la métrique QoE.

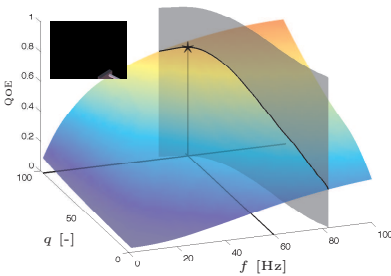
Fig. 55: L'adaptation contextuelle modifie les paramètres d'encodage, la fréquence f et la quantification q , pour obtenir la meilleure QoE possible en fonction du contenu du flux vidéo. Les valeurs de variables contextuelles sont constantes (vitesse $s = 0.4 \frac{\text{rad}}{\text{s}}$ et délai $d = 10 \text{ ms}$) et l'entrée unique est fixe ($u = 1$).



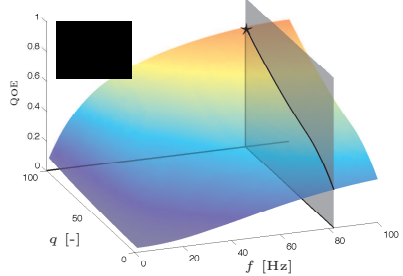
(a) Espace d'encodage augmenté à l'instant ❶.



(b) Espace d'encodage augmenté à l'instant ❷.

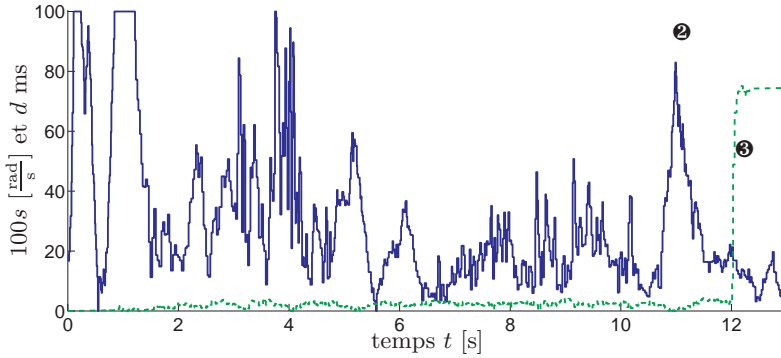


(c) Espace d'encodage augmenté à l'instant ❸.

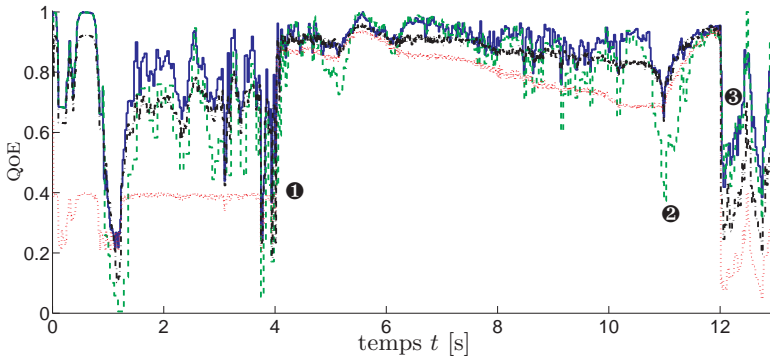


(d) Espace d'encodage augmenté à l'instant ❹.

Fig. 56: Le modèle contextuel change lorsque le contenu du flux vidéo évolue. Pour ce scénario, le contexte, et donc le modèle contextuel, est supposé constant. La combinaison optimale (\star) des paramètres d'encodage maximise la QoE.



(a) Evolution de la vitesse de la tête s (bleu continu avec un facteur de zoom de 100) et du délai global d (vert traitillé).



(b) Evolution de la QoE obtenue à partir du modèle, pour quatre scénarios : adaptation contextuelle (bleu continu), quantification q maximale (vert traitillé), fréquence f maximale (rouge pointillé), paramètres d'encodage f et q de valeurs similaires (noir mixte).

Fig. 57: Comparaison de la métrique de QoE pour différents scénarios.

5.6 Conclusion

Dans ce chapitre, nous avons présenté le schéma d'adaptation global qui permet d'obtenir la meilleure QoE possible en temps réel dans un contexte donné. Ce schéma combine les éléments présentés dans les chapitres précédents et l'adaptation contextuelle qui détermine les paramètres d'encodage optimaux. Il est implémenté au niveau du serveur sur une architecture client-serveur, mais utilise des informations issues du client.

Le contrôle de débit (Chap. 3) modifie la valeur de l'entrée unique u pour suivre le débit de référence donné par l'adaptation de transmission. La méthode du *mapping* (Chap. 4) convertit la valeur de l'entrée unique u en un ensemble de combinaisons de paramètres d'encodage représenté par la fonction réalisable g . L'adaptation contextuelle détermine la combinaison optimale en fonction du contexte, grâce au modèle de QoE (Chap. 2).

Les simulations et les mesures expérimentales ont permis de montrer le fonctionnement de l'adaptation contextuelle ainsi que de valider le schéma d'adaptation. Les résultats obtenus sont liés au profil choisi des variables contextuelles, au contenu du flux vidéo et au modèle de QoE. Cependant, le schéma d'adaptation global avec adaptation contextuelle des paramètres d'encodage permettait toujours d'obtenir la meilleure QoE possible.

Le schéma d'adaptation proposé se distingue des principales solutions d'adaptation [61–63] par l'approche générique, indépendante des choix technologiques. Notons que l'adaptation au niveau de la transmission [64–67] n'est pas considérée dans cette thèse ; uniquement un contrôle de débit du flux vidéo est proposé. Les travaux de Cranley et al. [68, 69] sur l'adaptation à la qualité d'un flux de *streaming* pourrait être étendue grâce au contrôle de débit combiné à la méthode du *mapping*. Cela permettrait de contrôler plus précisément le débit du flux vidéo, tout en considérant sa qualité.

Le schéma d'adaptation proposé étant générique, il est théoriquement possible de remplacer le modèle de QoE du Chap. 2 par un modèle issu de la littérature, et le contrôle de débit du Chap. 3 par

une stratégie équivalente. L'approche globale proposée ne se limite pas aux éléments développés dans le cadre de cette thèse, et pourra ainsi être utilisée avec de futures méthodes.

En prenant un point de vue plus proche du domaine de l'automatique, l'adaptation contextuelle de l'encodage peut être vue comme une solution de suractionnement. Le suractionnement concerne un système lorsqu'il a plus d'entrées qu'il n'en faut pour contrôler ses sorties. Les degrés de liberté laissés par le suractionnement sont généralement utilisés pour atteindre un objectif secondaire. Le contrôle de débit du flux vidéo ne nécessiterait qu'un paramètre d'encodage ; comme plusieurs paramètres d'encodage sont pris en compte dans notre schéma d'adaptation, le degré de liberté est utilisé pour maximiser la QoE au travers de l'adaptation contextuelle.

Conclusion

Cette thèse présente une méthode permettant d'adapter les paramètres d'encodage afin d'obtenir la meilleure expérience utilisateur possible en fonction d'un contexte dynamique.

La notion d'expérience utilisateur étant subjective, nous avons conçu la qualité d'expérience comme métrique objective et un modèle générique qui permet de prédire la qualité d'expérience en temps réel. Ce modèle est représenté par une fonction spline multivariable, qui peut être identifiée ou définie de manière générique.

Pour garantir une transmission efficace, le débit du flux vidéo de réalité augmentée est contrôlé en fonction de la bande passante en adaptant les différents paramètres d'encodage. Nous proposons une stratégie de contrôle de débit capable de gérer plusieurs paramètres d'encodage. L'encodeur est modélisé de manière abstraite sous forme de système linéaire monovariable, où la variation du contenu est traitée comme une perturbation. Un contrôleur stable et performant est développé pour le modèle abstrait.

Pour pouvoir implanter concrètement ce contrôleur, il est nécessaire de déterminer les combinaisons de paramètres de l'encodeur réel qui correspondent à l'entrée unique du modèle abstrait. Afin de trouver cette correspondance en temps réel, nous proposons un

nouvel algorithme direct basé sur un *mapping*. L'algorithme a été déterminé pour les encodeurs basés sur la transformée en cosinus discrète.

L'adaptation contextuelle choisit ensuite la combinaison de paramètres d'encodage qui maximise la qualité d'expérience grâce au modèle de qualité d'expérience développé précédemment. Finalement, le schéma d'adaptation global intègre le contrôle de débit, la méthode de *mapping* et l'adaptation contextuelle pour une implémentation en temps réel. Par exemple, lorsque l'utilisateur bouge sa tête rapidement, l'image est plus fortement compressée pour permettre d'augmenter la fréquence d'affichage et ainsi d'obtenir une meilleure perception du mouvement.

Ainsi, l'adaptation proposée permet d'offrir à l'utilisateur la meilleure expérience utilisateur possible dans un contexte donné.

6.1 Contributions

Cette thèse montre qu'une stratégie d'adaptation globale contribue à améliorer l'expérience utilisateur dans un domaine de l'interaction entre l'homme et la machine. Notre schéma d'adaptation offre une solution générique et indépendante des choix technologiques. Pour ce faire, nous proposons différents éléments.

1. **Conception d'un modèle générique pour la qualité d'expérience.** Le modèle objectif proposé dans le Chap. 2 permet de prédire la qualité d'expérience en temps réel en ne mesurant que les valeurs des paramètres d'encodage ainsi que des variables contextuelles. La procédure d'identification du modèle permet de l'adapter à différentes conditions d'utilisation, pour la réalité augmentée ou pour d'autres domaines.
2. **Représentation du modèle de qualité d'expérience par une fonction mathématique.** Le modèle est défini par une fonction spline multivariable (Sect. 2.3). Cette représentation générique permet d'étendre le modèle en y intégrant de nouvelles

variables. Ce modèle peut être projeté dans l'espace d'encodage pour ne représenter plus que l'effet des variables qui peuvent être adaptées en temps réel. Ainsi, le modèle peut facilement être utilisé pour adapter ces variables afin d'atteindre un certain objectif, en tenant compte des contraintes, comme par exemple de bande passante.

3. **Stratégie de contrôle de débit basée sur une approche système.** Afin de pouvoir contrôler le débit du flux vidéo en adaptant simultanément plusieurs paramètres d'encodage, un modèle abstrait de l'encodeur est proposé dans le Chap. 3. Ce modèle linéaire monovariable permet de concevoir un contrôleur de façon abstraite, pour atteindre une performance désirée. Ce modèle peut être utilisé pour développer des stratégies de contrôle adaptées à l'encodage de flux temps réel qui ne sont pas forcément liés à la réalité augmentée.
4. **Algorithme direct de correspondance entre le modèle générique et un encodeur.** Pour pouvoir appliquer le contrôle de débit conçu sur le modèle abstrait, la méthode de *mapping* est proposée dans le Chap. 4. Un nouvel algorithme direct a été défini pour obtenir ce *mapping* en temps réel (Sect. 4.5).
5. **Schéma d'adaptation global.** Le schéma d'adaptation proposé dans le Chap. 5 intègre l'adaptation de la transmission, le contrôle de débit et l'adaptation contextuelle. Des simulations et des mesures expérimentales permettent de valider le schéma d'adaptation global. La structure proposée permet d'obtenir la meilleure qualité d'expérience possible, sans toucher à la conception ou à l'architecture de l'application d'AR. Cette structure générique permet de remplacer ou d'améliorer l'un ou l'autre des éléments de l'adaptation.

6.2 Perspectives

Cette thèse apporte plusieurs solutions dans le domaine de l'adaptation pour l'expérience utilisateur, mais elle pose également de nouveaux défis. Nous présentons ici quelques uns de ces défis.

6.2.1 Réalité augmentée

Afin d'être détaché des choix technologiques, nous avons utilisé une approche générique. Cela donne à cette thèse des perspectives pour l'avenir, en suivant l'évolution de la réalité augmentée.

L'évolution de la réalité augmentée est grandement influencée par les avancées techniques dans différents domaines, comme par exemple la puissance de calcul des appareils mobiles, la performance des systèmes de communication, l'utilisabilité des périphériques d'affichage et la précision des système de positionnement avec prédiction. Ces améliorations sont nécessaires pour que la réalité augmentée se développe hors du domaine de la recherche.

Cette évolution sera à prendre en considération pour le développement de nouvelles solutions d'adaptation, basées sur la méthodologie proposée. Par exemple, si un nouvel encodeur spécialement approprié pour la réalité augmentée venait à s'imposer, il serait nécessaire d'évaluer son impact sur l'expérience utilisateur. De même, il faudrait modifier le contrôle de débit en tenant compte des particularités de ce nouvel encodeur.

6.2.2 Expérience utilisateur

Le domaine de l'expérience utilisateur est souvent plus vaste et complexe que nous l'imaginons. L'aspect subjectif de l'utilisation de nouvelles technologies est tel, qu'il semble difficile de modéliser globalement l'expérience utilisateur. Dans cette thèse, nous n'avons pris en compte que certaines variables qui influencent l'expérience utilisateur dans le domaine de la réalité augmentée. Voici quelques améliorations envisageables :

- Pour pouvoir utiliser l'adaptation à large échelle, il est conseillé d'identifier le modèle de qualité d'expérience plus précisément. Cependant, l'évaluation subjective correspond à un domaine de recherche en soi. De plus amples évaluations seraient nécessaires pour quantifier la sensibilité des utilisateurs par rapport aux différentes variables prises en compte et pour évaluer

la validité du modèle pour plusieurs utilisateurs ou catégories d'utilisateurs.

- Il serait intéressant d'identifier comment l'utilisateur réagit lorsqu'une variable (par exemple la fréquence) varie constamment ou brusquement. Cette notion de variation pourrait ainsi être intégrée dans le schéma d'adaptation pour modérer les variations des paramètres d'encodage. D'autres variables ou éléments (par exemple la dynamique de la scène virtuelle) influencent l'expérience utilisateur. Nous avons par exemple négligé l'effet de la variation de variables.
- Dans une approche intégrant la transmission, il semble judicieux de quantifier comment le canal de communication influence l'expérience utilisateur par la transformation du flux vidéo. L'impact des pertes de paquets ou de la gigue pourraient ainsi être analysés. Cela permettrait d'en tenir compte au niveau de l'adaptation de transmission.

6.2.3 Encodage de flux vidéo

L'encodage de flux vidéo n'est évidemment pas spécifique à la réalité augmentée. De ce fait, les contributions présentées dans cette thèse et les nouveautés dans le domaine de l'encodage peuvent être combinées.

- Des encodeurs avec compression temporelle ou orientés pour le mouvement, par exemple MPEG, pourraient remplacer avantageusement l'encodeur M-JPEG que nous avons utilisé pour compresser le flux vidéo de réalité augmentée. Comme l'encodeur MPEG est également basé sur la transformée en cosinus discrète, le modèle linéaire de l'encodeur resterait valable. Cependant, il y a un plus grand nombre de paramètres d'encodage à gérer ; l'évaluation du modèle de qualité d'expérience et l'algorithme direct de *mapping* pourraient présenter un degré de complexité supérieur. Mais le taux de compression et la qualité du flux encodé seraient certainement améliorés.

- Le contrôle de débit proposé dans cette thèse pourrait être utilisé pour d'autres flux vidéos que la réalité augmentée. Comme notre stratégie est développée pour l'encodage de flux temps réels, elle peut être envisagée pour encoder des flux de vidéo-conférences par exemple.
- La méthode du *mapping* a été présentée pour un encodeur ayant deux paramètres d'encodage. Mais comme cette méthode est générique, elle peut être utilisée pour un plus grand nombre de paramètres d'encodage, par exemple en utilisant plusieurs paramètres de la compression MPEG. Le graphique à quadrants devrait être étendu à des dimensions supérieures pour représenter le *mapping*, mais le concept reste similaire.
- La fonction réalisable g représente les combinaisons de paramètres d'encodage qui correspondent à une variable liée au débit du flux vidéo. La fonction réalisable est donc liée à une contrainte, ici la contrainte de débit. Il est possible de tenir compte de contraintes différentes, par exemple une contrainte de temps d'utilisation des ressources du processeur, car il dépend d'un ou plusieurs paramètres d'encodage. Cela modifierait considérablement le *mapping* qui devrait être réévalué.

6.2.4 Adaptation

Le schéma d'adaptation proposé peut être étendu, par exemple pour intégrer l'adaptation de transmission, ou pour utiliser l'appréciation de l'utilisateur en temps réel afin d'adapter le modèle de qualité d'expérience. La structure proposée pour l'adaptation permet d'intégrer ce type d'extension.

FIN

A

Fonctions splines

Ce bref rappel sur les splines introduit les bases nécessaires à la compréhension du modèle de QoE. La coupe d'une spline vers un espace de dimension inférieure présente comment passer du modèle $M(v_e, v_c)$ au modèle contextuel $M_c(v_e, k)$. Le modèle M est monotone dans certaines directions ; lorsque la variable augmente, la métrique QoE augmente également. Afin d'intégrer ce comportement monotone, la notion de monotonie d'une spline est expliquée.

Les fonctions *splines* sont utilisées pour les problèmes d'approximation, que ce soit pour l'interpolation (la courbe passe par les points de mesure) ou pour le lissage (la courbe présente une allure lisse mais ne passe pas forcément par les points de mesure). Dans ce contexte, les splines sont souvent préférées aux polynômes car, étant définies par morceaux, il est possible d'obtenir des résultats similaires avec des fonctions d'ordres inférieurs et d'éviter le phénomène de Runge (notion d'instabilité quand le nombre de points change) [71].

A.1 Splines monovariabiles

La spline monovariabie $S(x)$ est une fonction polynomiale définie par morceaux sur un intervalle donné.

$$\begin{aligned} S : x &\mapsto y = S(x) \\ [a, b] &\mapsto \mathbb{R} \end{aligned}$$

L'intervalle $[a, b]$ est découpé en sous-intervalles $[\tau_i, \tau_{i+1}]$, tel que $a = \tau_1 \leq \tau_2 \leq \dots \leq \tau_{p+1} = b$. Les scalaires τ_i sont appelés les *noeuds*. Le vecteur $\tau = [\tau_1, \dots, \tau_{p+1}]$ est appelé séquence de noeuds. Des noeuds successifs peuvent avoir la même valeur, on parlera alors de noeuds multiples.

Soit le polynôme $P(x)$ d'ordre w (ou de degré $w - 1$) :

$$P(x) = \sum_{j=0}^{w-1} \alpha_j x^j$$

avec les coefficients du polynôme α .

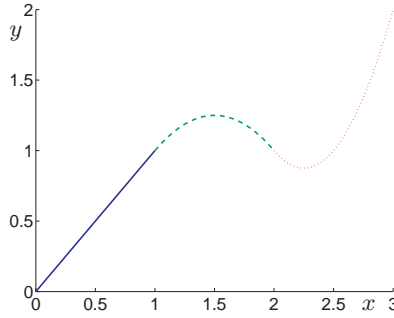
La spline $S(x)$ est définie par les polynômes $P_i(x)$.

$$S(x) = \begin{cases} P_1(x), & \tau_1 \leq x < \tau_2 \\ \dots \\ P_i(x), & \tau_i \leq x < \tau_{i+1} \\ \dots \\ P_p(x), & \tau_p \leq x < \tau_{p+1} \end{cases}$$

Une spline est dite d'ordre w lorsque ses polynômes sont d'ordre w et que la spline et ses dérivées jusqu'à l'ordre $w - 2$ sont continues.

$$S \in C^{w-2}[a, b]$$

Prenons à titre d'exemple la spline $S(x)$ d'ordre $w = 3$ (donc de degré 2) définie dans l'intervalle $[0, 3]$ (Fig. 1) qui passe par les points $(x, y) : (0, 0), (1, 1), (2, 1), (3, 2)$.

Fig. 1: Une spline monovariabiles d'ordre $w = 3$.

$$S(x) = \begin{cases} P_1(x), & 0 \leq x \leq 1 \\ P_2(x), & 1 \leq x \leq 2 \\ P_3(x), & 2 \leq x \leq 3 \end{cases}$$

$$P_1(x) = x$$

$$P_2(x) = -x^2 + 3x - 1$$

$$P_3(x) = 2x^2 - 9x + 11$$

où

$$P_2(1) = P_1(1) = 1$$

$$P_3(2) = P_2(2) = 1$$

et les dérivées jusqu'à l'ordre $w - 2 = 1$

$$\frac{d}{dx}P_2(x)\Big|_{x=1} = \frac{d}{dx}P_1(x)\Big|_{x=1} = 1$$

$$\frac{d}{dx}P_3(x)\Big|_{x=2} = \frac{d}{dx}P_2(x)\Big|_{x=2} = -1.$$

A.2 B-splines monovariabiles

Introduites par Curry et Schoenberg, les *B-splines* ("B" pour base) forment une base pour l'espace des splines avec une séquence de noeuds donnée [72]. Une spline $S_w(x)$ d'ordre w est exprimée de manière unique comme une combinaison linéaire de N_B B-splines $B_{j,w}(x)$ d'ordre w sous la forme :

$$S_w(x) = \sum_{j=1}^{N_B} \alpha_j B_{j,w}(x) \quad (1)$$

avec les coefficients α de la spline, et où les B-splines (normalisées) sont définies grâce à la formule récursive de Carl de Boor [73] par :

$$B_{j,1}(x) = \begin{cases} 1, & \tau_j \leq x \leq \tau_{j+1} \\ 0, & \text{sinon} \end{cases} \quad (2)$$

$$B_{j,w}(x) = \frac{x - \tau_j}{\tau_{j+w-1} - \tau_j} B_{j,w-1}(x) + \frac{\tau_{j+w} - x}{\tau_{j+w} - \tau_{j+1}} B_{j+1,w-1}(x). \quad (3)$$

Quelques points importants concernant les B-splines :

- Une B-spline est une spline positive à support minimal (le coefficient unique vaut 1).
- Une B-spline d'ordre w est définie de manière unique pour $w + 1$ noeuds.
- Entre chaque paire de noeuds successifs, une B-spline d'ordre w est un polynôme d'ordre w .
- De 1 à w noeuds peuvent être présents aux bornes ; une multiplicité de n permet que les dérivées d'ordre $w - n$ soient non nulles.
- Pour une séquence de m noeuds donnés, la base des splines comporte $l = m - w$ B-splines d'ordre w .

Résumé

Une spline est entièrement définie par sa séquence de noeuds τ et ses coefficients α .

$$S(x) = \sum_{j=1}^{N_B} \alpha_j B_j(x) \quad (4)$$

Afin d'améliorer la lisibilité, l'indice w indiquant l'ordre d'une fonction est désormais omis.

Lorsqu'une spline est utilisée pour interpoler ou lisser des points de mesure, les coefficients α sont déterminés par optimisation, tels que la spline résultante passe au plus près des points de mesure. La Sect. 2.4.2 présente l'identification des coefficients α à partir de mesures. Cependant, les coefficients α peuvent aussi être choisis arbitrairement pour définir une spline.

Avant de passer à un nombre supérieur de variables, il est utile de représenter graphiquement la construction d'une spline monovariante.

A.3 Construction d'une spline monovariante

Soit la spline définie par la séquence de $m = 8$ noeuds $\tau = [0, 1, 2, 3, 4, 5, 5, 5]$ (avec un noeud d'une multiplicité triple en 5) et les $l = 5$ coefficients $\alpha = [1.5, 0.5, 1, 0.5, 0.5]$. Il en découle que l'ordre de la spline est $w = m - l = 3$; il s'agit d'une fonction quadratique. La base est composée de $l = 5$ B-splines d'ordre $w = 3$ (Fig. 2a). Chacune des B-splines est définie pour les séquences de $w + 1 = 4$ noeuds successifs $[0, 1, 2, 3]$, $[1, 2, 3, 4]$, \dots , $[4, 5, 5, 5]$ et leur coefficient 1. La spline engendrée dans cette base correspond à la somme des $l = 5$ B-splines pondérées par les coefficients α (Fig. 2b). La multiplicité triple à l'abscisse 5 permet à cette spline d'avoir une pente et une valeur non nulles.

A.4 Construction d'une base multivariable

Les splines s'appliquent aussi pour plusieurs variables [74]. C'est le cas par exemple en 3D pour définir des surfaces dans la conception

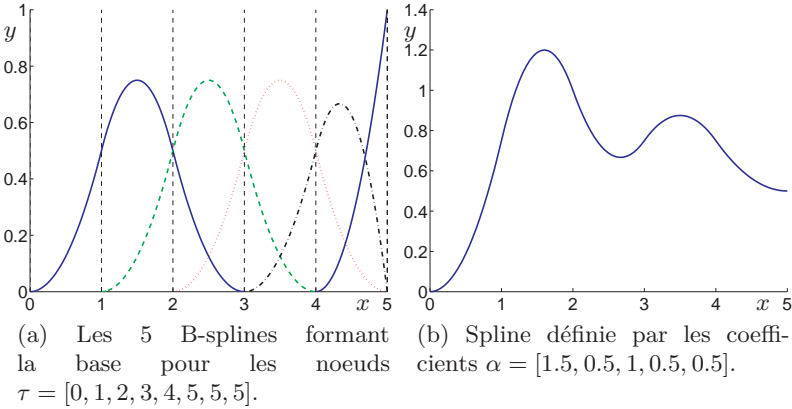


Fig. 2: Construction d'une spline à partir des noeuds et des coefficients.

de pièces mécaniques ou l'affichage d'objets virtuels. Il est également possible d'utiliser les splines pour représenter une hypersurface comportant un nombre encore plus élevé de variables.

La B-spline à N variables $B_{(\zeta_1, \dots, \zeta_N)}(x_1, \dots, x_N)$ correspond à une extension de la B-spline monovariante. Une B-spline multivariante est une fonction séparable issue du produit tensoriel de N B-splines monovariantes $B_{\zeta_i}(x_i)$, où l'indice i indique la variable correspondante et l'indice ζ_i indique la B-spline monovariante sélectionnée pour la variable correspondante.

$$B : (x_1, \dots, x_N) \mapsto y = B_{(\zeta_1, \dots, \zeta_N)}(x_1, \dots, x_N) \quad (5)$$

$$\mathbb{R}^N \mapsto \mathbb{R} \quad (6)$$

$$B_{(\zeta_1, \dots, \zeta_N)}(x_1, \dots, x_N) = \prod_{i=1}^N B_{\zeta_i}(x_i) \quad (7)$$

Une B-spline multivariable est définie par un ensemble de N séquences de noeuds τ_i (ce sont des vecteurs et non plus des scalaires). Comme pour le cas monovariante, les B-splines multivariées forment une base pour l'espace des splines à N variables avec un ensemble de séquences de noeuds données. Il y a m_i noeuds dans la séquence τ_i et l'ordre de la B-spline correspondante est w_i . Le nombre N_B de B-splines qui composent la base dépend du nombre N de variables et du nombre de B-splines monovariées pour chacune des variables.

$$N_B = \prod_{i=1}^N l_i$$

$$l_i = m_i - w_i$$

A titre d'exemple, la base des splines d'ordre $w = w_i = 3$ à $N = 2$ variables (x_1, x_2) peut être représentée dans l'espace 3D (Fig. 3). Avec un ensemble de séquences de noeuds $\tau = \{\tau_1, \tau_2\} = \{[0, 1, 2, 3, 4, 5, 5, 5], [10, 10, 10, 13, 19, 19, 19]\}$, la base contient $N_B = 20$ B-splines multivariées $B_{(\zeta_1, \zeta_2)}(x_1, x_2)$ issues des produits tensoriels de deux B-splines monovariées $B_{\zeta_i}(x_i)$.

$$B_{(\zeta_1, \zeta_2)}(x_1, x_2) = B_{\zeta_1}(x_1)B_{\zeta_2}(x_2)$$

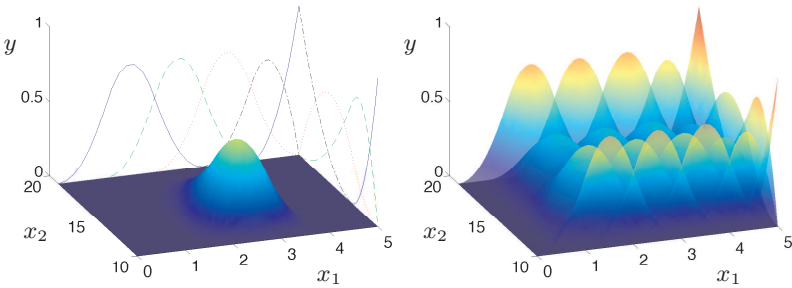
$$\zeta_i = \{1, \dots, l_i\}$$

$$i = \{1, 2\}$$

Définition 14 (Spline multivariable). *La spline multivariable S_N est entièrement définie avec son ensemble de séquences de noeuds τ et le tableau de coefficients α , de dimension N (il y a N_B coefficients).*

$$\begin{aligned}
 S_N &= S(x_1, \dots, x_N) \\
 &= \sum_{\zeta_1=1}^{l_1} \dots \sum_{\zeta_N=1}^{l_N} \alpha_{(\zeta_1, \dots, \zeta_N)} B_{(\zeta_1, \dots, \zeta_N)}(x_1, \dots, x_N) \\
 &= \sum_{\zeta_1=1}^{l_1} \dots \sum_{\zeta_N=1}^{l_N} \alpha_{(\zeta_1, \dots, \zeta_N)} \prod_{i=1}^N B_{\zeta_i}(x_i)
 \end{aligned} \tag{8}$$

où les $B_{(\zeta_1, \dots, \zeta_N)}$ correspondent aux B-splines multivariées issues du produit des B-splines monovariées $B_{\zeta_i}(x_i)$.



(a) Les B-splines monovariées sont affichées sur leur axe correspondant. La B-spline multivariée correspond au produit tensoriel de deux B-splines monovariées $\tau = \{[1, 2, 3, 4], [10, 10, 13, 19]\}$.

(b) Ensemble des 20 B-splines composant la base. On remarque que dans la première dimension, les B-splines partent de zéro avec une pente nulle ; ceci est dû au fait qu'il n'y ait pas de multiplicité pour les noeuds en 0.

Fig. 3: Illustration de la base des splines d'ordre 3 pour un ensemble de séquences de noeuds donnés.

A.5 Coupe vers un espace de dimension inférieure

Grâce au fait que les B-splines sont des fonctions séparables, une spline S_N à N variables peut être coupé vers un espace de dimension inférieure à R variables pour obtenir une section S_R . La troncation des $N - R$ variables est issue de l'intersection de la spline S_N avec un hyperplan, de façon analogue aux coniques issues de l'intersection d'un cône avec un plan.

Soit la fonction à N variables :

$$S_N = S(x_1, \dots, x_N)$$

qu'on veut couper vers l'espace à R variables (x_1, \dots, x_R) pour obtenir la fonction :

$$S_R = S(x_1, \dots, x_R, \bar{x}_{R+1}, \dots, \bar{x}_N)$$

où les $N - R$ paramètres \bar{x}_i ne sont plus des variables mais ont des constantes. La fonction S_N peut s'écrire en dissociant les variables x_1 à x_R des variables x_{R+1} à x_N .

$$\begin{aligned} S_N &= S(x_1, \dots, x_R, x_{R+1}, \dots, x_N) \\ &= \sum_{\zeta_1=1}^{l_1} \dots \sum_{\zeta_N=1}^{l_N} \alpha_{(\zeta_1, \dots, \zeta_N)} \prod_{i=1}^R B_{\zeta_i}(x_i) \prod_{i=R+1}^N B_{\zeta_i}(x_i) \end{aligned}$$

De manière similaire,

$$\begin{aligned} S_R &= S(x_1, \dots, x_R) \\ &= \sum_{\zeta_1=1}^{l_1} \dots \sum_{\zeta_R=1}^{l_R} \alpha'_{(\zeta_1, \dots, \zeta_R)} \prod_{i=1}^R B_{\zeta_i}(x_i) \end{aligned}$$

où α' correspond au nouveau tableau de coefficients coupé vers R dimensions. Il peut être trouvé en développant les précédentes équations et en utilisant le fait que les \bar{x}_j sont fixes et connus.

$$\begin{aligned}
 S_R &= S(x_1, \dots, x_R, \bar{x}_{R+1}, \dots, \bar{x}_N) \\
 &= \sum_{\zeta_1=1}^{l_1} \dots \sum_{\zeta_R=1}^{l_R} \underbrace{\sum_{\zeta_{R+1}=1}^{l_{R+1}} \dots \sum_{\zeta_N=1}^{l_N} \alpha_{(\zeta_1, \dots, \zeta_N)}}_{\alpha'_{(\zeta_1, \dots, \zeta_R)}} \prod_{i=R+1}^N B_{\zeta_i}(\bar{x}_i) \prod_{i=1}^R B_{\zeta_i}(x_i)
 \end{aligned} \tag{9}$$

La nouvelle spline S_R est maintenant définie par le tableau de coefficients α' déterminé par l'Equ. (9) et la séquence de séquences de noeuds τ' correspondant aux dimensions restantes de la séquence initiale τ .

Prenons un exemple. Soit la spline $S(x_1, x_2)$ d'ordre $w = 3$ à $N = 2$ variables, définie par les noeuds τ et la matrice (tableau de dimension $N = 2$) coefficients α suivants (Fig. 4) :

$$\begin{aligned}
 \tau &= \{[0, 1, 2, 3, 4, 5, 5, 5], [10, 10, 10, 13, 19, 19, 19]\} \\
 \alpha &= \begin{pmatrix} 0 & 1 & 2 & 0 & 1 \\ 1 & 0 & 2 & 1 & 3 \\ 2 & 2 & 3 & 1 & 2 \\ 1 & 2 & 2 & 3 & 2 \end{pmatrix}.
 \end{aligned}$$

En suivant la démarche présentée précédemment, il est possible de déterminer la spline monovariable $S(x_1)$ correspondant à la coupe de la spline multivariable $S(x_1, x_2)$ dans le plan (x_1, y) pour n'importe quelle valeur de \bar{x}_2 . A titre d'exemple, nous présentons le développement pour trouver la séquence des noeuds τ' et le vecteur des coefficients α' de la spline pour la valeur $\bar{x}_2 = 13$. Les noeuds correspondent simplement aux noeuds de la dimension restante $\tau' = \tau_1 = [0, 1, 2, 3, 4, 5, 5, 5]$. Les 5 coefficients α' sont calculés à partir de la matrice de coefficients α et de la valeur des B-splines monovariables en $\bar{x}_2 = 13$. Les valeurs de ces 4 B-splines $B_i(\bar{x}_2)$ valent respectivement $0, \frac{2}{3}, \frac{1}{3}$ et 0 . Les coefficients $\alpha'_{(\zeta_1, \dots, \zeta_R)}$ sont calculés de manière générique.

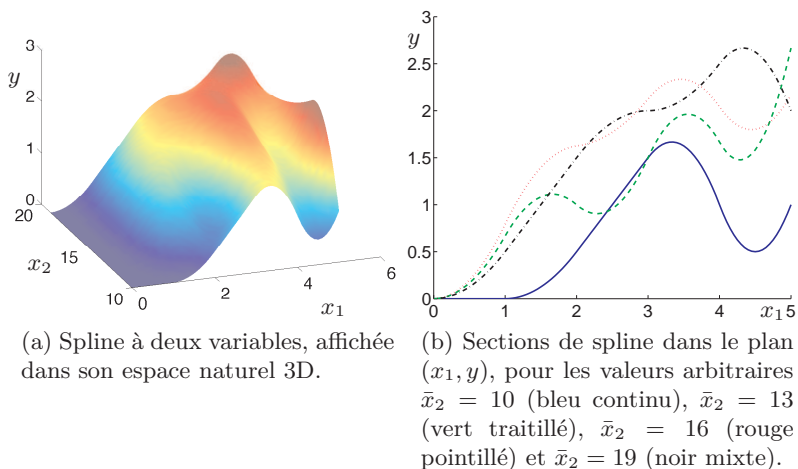


Fig. 4: Spline S à deux variables.

$$\alpha'_{(\zeta_1, \dots, \zeta_R)} = \sum_{\zeta_{R+1}=1}^{l_{R+1}} \dots \sum_{\zeta_N=1}^{l_N} \alpha_{(\zeta_1, \dots, \zeta_N)} \prod_{i=R+1}^N B_{\zeta_i}(\bar{x}_i) \quad (10)$$

Dans l'exemple, la spline à deux variables est coupée vers l'espace de x_1 . Les coefficients α'_{ζ_1} sont donnés par :

$$\begin{aligned} \alpha'_{\zeta_1} &= \sum_{\zeta_2=1}^4 \alpha_{(\zeta_1, \zeta_2)} \prod_{i=2}^2 B_{\zeta_i}(\bar{x}_2) \\ &= \sum_{\zeta_2=1}^4 \alpha_{(\zeta_1, \zeta_2)} B_{\zeta_2}(\bar{x}_2). \end{aligned} \quad (11)$$

Ainsi, le vecteur $\alpha' = [\alpha'_1, \alpha'_2, \alpha'_3, \alpha'_4]$ devient :

$$\begin{aligned}
 \alpha' &= \alpha_{(\bullet,1)}B_1(\bar{x}_2) + \alpha_{(\bullet,2)}B_2(\bar{x}_2) + \alpha_{(\bullet,3)}B_3(\bar{x}_2) + \alpha_{(\bullet,4)}B_4(\bar{x}_2) \\
 &= 0[0, 1, 2, 0, 1] + \frac{2}{3}[1, 0, 2, 1, 3] + \frac{1}{3}[2, 2, 3, 1, 2] + 0[1, 2, 2, 3, 2] \\
 &= \frac{1}{3}[4, 2, 7, 3, 8]
 \end{aligned}
 \tag{12}$$

où la notation $\alpha_{(\bullet,i)}$ correspond au vecteur de toutes les lignes mais seulement la i^e colonne de la matrice α .

La spline définie par τ' et α' est représentée dans la Fig. 4b, avec trois autres sections de spline pour des valeurs de \bar{x}_2 différentes.

A.6 Monotonie

Les coefficients d'une spline ont des propriétés intéressantes [75]. Il n'y a pas plus de changements de signe dans la spline S qu'il n'y en a dans la séquence de coefficients α . De même, si tous les coefficients α_i sont non-négatifs, alors la spline S est non-négative. Et enfin, si les coefficients α sont monotones dans la direction d'une variable x_j , alors la spline est monotone dans cette direction. Ainsi, une spline est monotone dans toutes les directions si les coefficients sont monotones dans toutes les directions. Notez que nous utilisons ici le terme "spline monotone" lors que les dérivées partielles de ces directions ne changent pas de signe.

Pour la séquence de noeuds identique au cas précédent, la spline multivariable est monotone pour les coefficients α' suivants (Fig. 5) :

$$\alpha' = \begin{array}{c} \overrightarrow{\left(\begin{array}{cccc} 0 & 0 & 1 & 1 & 2 \\ 0 & 1 & 1 & 2 & 2 \\ 1 & 2 & 2 & 2 & 3 \\ 2 & 3 & 3 & 3 & 3 \end{array} \right)} \\ \downarrow \end{array}$$

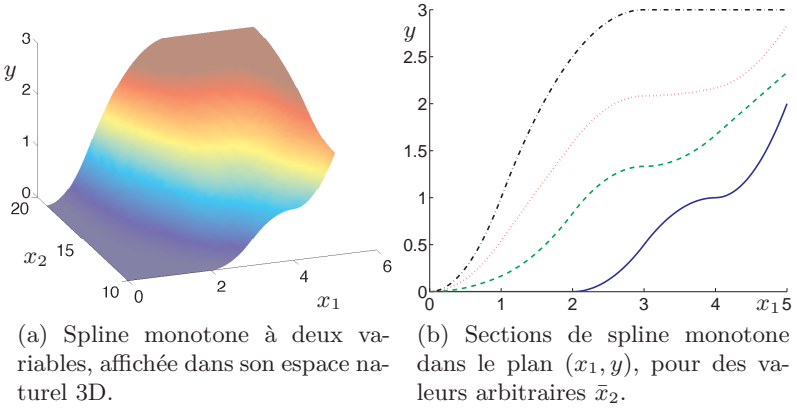


Fig. 5: Spline multivariable monotone.

B

Noeuds et coefficients du modèle générique

Le modèle générique de QoE $M_g(f, q, d, s)$ est utilisé pour différentes simulations. Il est défini avec les noeuds τ et les coefficients α choisis arbitrairement pour illustrer l'allure générale du modèle.

$$\begin{aligned}\tau &= \{\tau_f, \tau_q, \tau_d, \tau_s\} \\ \tau_f &= [1, 1, 1, 10, 25, 50, 100, 100, 100] \\ \tau_q &= [0, 0, 0, 30, 70, 100, 100, 100] \\ \tau_d &= [0, 0, 0, 50, 100, 100, 100] \\ \tau_s &= [0, 0, 0, 0.3, 0.6, 1, 1, 1]\end{aligned}$$

$$\alpha_{(\bullet, \bullet, 1, 1)} = \begin{bmatrix} 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \end{bmatrix} \quad \alpha_{(\bullet, \bullet, 2, 1)} = \begin{bmatrix} 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \end{bmatrix}$$

$$\alpha_{(\bullet, \bullet, 3, 1)} = \begin{bmatrix} 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \end{bmatrix} \quad \alpha_{(\bullet, \bullet, 4, 1)} = \begin{bmatrix} 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \end{bmatrix}$$

$$\alpha_{(\bullet, \bullet, 1, 2)} = \begin{bmatrix} 0.2 & 0.2 & 0.3 & 0.4 & 0.5 \\ 0.2 & 0.4 & 0.5 & 0.6 & 0.7 \\ 0.3 & 0.4 & 0.6 & 0.8 & 0.8 \\ 0.4 & 0.5 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \\ 0.4 & 0.6 & 0.7 & 0.9 & 1.0 \end{bmatrix} \quad \alpha_{(\bullet, \bullet, 2, 2)} = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.1 & 0.2 \\ 0.0 & 0.0 & 0.1 & 0.2 & 0.3 \\ 0.0 & 0.1 & 0.2 & 0.4 & 0.4 \\ 0.0 & 0.2 & 0.3 & 0.5 & 0.6 \\ 0.0 & 0.2 & 0.3 & 0.5 & 0.6 \\ 0.1 & 0.2 & 0.3 & 0.5 & 0.6 \end{bmatrix}$$

$$\alpha_{(\bullet, \bullet, 3, 2)} = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.1 \\ 0.0 & 0.0 & 0.0 & 0.2 & 0.2 \\ 0.0 & 0.0 & 0.1 & 0.3 & 0.4 \\ 0.0 & 0.0 & 0.1 & 0.3 & 0.4 \\ 0.0 & 0.1 & 0.1 & 0.3 & 0.4 \end{bmatrix} \quad \alpha_{(\bullet, \bullet, 4, 2)} = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.1 & 0.2 \\ 0.0 & 0.0 & 0.0 & 0.1 & 0.2 \\ 0.0 & 0.0 & 0.0 & 0.1 & 0.2 \end{bmatrix}$$

C

Sources des images

Quatre images sont utilisées pour illustrer les caractéristiques d'encodage. Ces images, au format bitmap, ont une résolution de 800×600 pixels et une profondeur de couleurs de 24 bits.



La première image est issue de notre prototype de génération d'images pour l'AR.

La deuxième image provient de <http://www.scottishwidows.co.uk/tv/downloads/swtv2-800X600.bmp>.

La troisième image provient de <http://www.castleuk.net/fun/2006/images/alnwick800.bmp>.

La quatrième image provient de http://www.hongai.edu.hk/images/school_logo/hnc-800-600.bmp.

Références

1. Ulrich Engelke and Hans-Jürgen Zepernick. Perceptual-based quality metrics for image and video services: A survey. In *Proceedings of the 3rd EuroNGI Conference on Next Generation Internet Networks*, pages 190–197. IEEE, 2007.
2. Gheorghita Ghinea, Robert S. Fish, and Johnson P. Thomas. Using quality of perception for improved multimedia communication. In *Proceedings of the AFRICON*, volume 2, pages 1241–1246. IEEE, September 1999.
3. Gheorghita Ghinea, Johnson P. Thomas, and Robert S. Fish. Multimedia, network protocols and users - bridging the gap. In *Proceedings of the 7th International Conference on Multimedia (Part 1) (MULTIMEDIA 99)*, pages 473–476, New York, USA, 1999. ACM.
4. Stephen R. Gulliver and Gheorghita Ghinea. Changing frame rate, changing satisfaction? [multimedia quality of perception]. In *Proceedings of the International Conference on Multimedia and Expo (ICME 04)*, volume 1, pages 177–180. IEEE, June 2004.
5. Gheorghita Ghinea and Johnson P. Thomas. Quality of perception: User quality of service in multimedia presentations. *IEEE Transactions on Multimedia*, 7(4):786–789, August 2005.
6. Gheorghita Ghinea and Sherry Y. Chen. Perceived quality of multimedia educational content: A cognitive style approach. *Multimedia Systems*, 11(3):271–279, March 2006.

7. Sherry Y. Chen, Gheorghita Ghinea, and Robert D. Macredie. A cognitive approach to user perception of multimedia quality: An empirical investigation. *International Journal of Human-Computer Studies*, 64(12):1200–1213, 2006.
8. Fernando Pereira. A triple user characterization model for video adaptation and quality of experience evaluation. In *Proceedings of the 7th Workshop on Multimedia Signal Processing*, pages 1–4. IEEE, 2005.
9. Andrew Perkis, Solveig Munkeby, and Odd Inge Hillestad. A model for measuring Quality of Experience. In *Proceedings of the 7th Nordic Signal Processing Symposium (NORSIG 2006)*, pages 198–201. IEEE, 2006.
10. Hamid Rahid Sheikh and Alan C. Bovik. Image information and visual quality. *IEEE Transactions on Image Processing*, 15(2):430–444, February 2006.
11. Antony W. Rix, Alex Bourret, and Mike P. Hollier. Models of human perception. *BT Technology Journal*, 17(1):24–34, January 1999.
12. Stefan Winkler. *Vision models and quality metrics for image processing applications*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne, Lausanne, 2000.
13. Olivier Verscheure, Xavier Garcia, Gunnar Karlsson, and Jean-Pierre Hubaux. User-oriented QoS in packet video delivery. *IEEE Network*, 12(6):12–21, November 1998.
14. Olivier Verscheure. *User-oriented QoS in MPEG video delivery*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne, Lausanne, 1999.
15. Sandeep Kanumuri, Pamela C. Cosman, Amy R. Reibman, and Vinay A. Vaishampayan. Modeling packet-loss visibility in MPEG-2 video. *IEEE Transactions on Multimedia*, 8(2):341–355, April 2006.
16. Nicola Cranley, Perry Perry, and Liam Murphy. Optimum adaptation trajectories for streamed multimedia. *Multimedia Systems*, 10(5):392–401, August 2005.
17. Gayatri Yadavalli, Mark Masry, and Sheila S. Hemami. Frame rate preferences in low bit rate video. In *Proceedings of the International Conference on Image Processing (ICIP 2003)*, volume 1, pages I–441–I–444. IEEE, 2003.

18. H. Koumaras, A. Kourtis, D. Martakos, and J. Lauterjung. Quantified PQoS assessment based on fast estimation of the spatial and temporal activity level. *Multimedia Tools and Applications*, 34(3):355–374, September 2007.
19. Ronald T. Azuma. A survey of augmented reality. *Presence-Teleoperators and Virtual Environments*, 6(4):355–385, August 1997.
20. W. Pasman and F.W. Jansen. Comparing simplification and image-based techniques for 3D client-server rendering systems. *IEEE Transactions on Visualization and Computer Graphics*, 9(2):226–240, April 2003.
21. Stephen R. Ellis, Mark J. Young, Bernard D. Adelstein, and Sheryl M. Ehrlich. Discrimination of changes in latency during head movement. In *Proceedings of the 8th International Conference on Human-Computer Interaction (HCI 1999) on Human-Computer Interaction: Communication, Cooperation, and Application Design*, volume 2, pages 1129–1133. ACM, 1999.
22. Robert S. Allison, Laurence R. Harris, Michael Jenkin, Ursula Jasiodedzka, and James E. Zacher. Tolerance of temporal delay in virtual environments. In *Proceedings of the Virtual Reality Conference (VR 01)*, pages 247–254. IEEE, 2001.
23. Bernard D. Adelstein, Thomas G. Lee, and Stephen R. Ellis. Head tracking latency in virtual environments: Psychophysics and a model. In *Proceedings of the 47th Annual Meeting on Human Factors and Ergonomics Society*, pages 2083–2087. Human Factors and Ergonomics Society, 2003.
24. Katerina Mania, Bernard D. Adelstein, Stephen R. Ellis, and Michael I. Hill. Perceptual sensitivity to head tracking latency in virtual environments with varying degrees of scene complexity. In *Proceedings of the 1st Symposium on Applied Perception in Graphics and Visualization*, pages 39–47, Los Angeles, California, 2004. ACM.
25. Jason Jerald, Tabitha Peck, Frank Steinicke, and Mary Whitton. Sensitivity to scene motion for phases of head yaws. In *Proceedings of the 5th Symposium on Applied Perception in Graphics and Visualization*, pages 155–162, Los Angeles, California, 2008. ACM.

26. Woodrow Barfield and Claudia Hendrix. The effect of update rate on the sense of presence within virtual environments. *Virtual Reality*, 1(1):3–15, June 1995.
27. Whung Yee Lai and Henry Been-Lirn Duh. Effects of frame rate for visualization of dynamic quantitative information in a head-mounted display. In *Proceedings of the International Conference on Systems, Man and Cybernetics*, volume 7, pages 6485–6490. IEEE, 2004.
28. Damien Perritaz, Christophe Salzmann, and Denis Gillet. User perception model for wearable supervision systems. In *Proceedings of the 4th International Conference on Mobile Technology, Applications and Systems and the 1st International Symposium on Computer Human Interaction in Mobile Technology (Mobility 2007)*, Singapore, September 2007. ACM.
29. Gerda J.F. Smets and Kees J. Overbeeke. Trade-off between resolution and interactivity in spatial task performance. *IEEE Computer Graphics and Applications*, 15(5):46–51, September 1995.
30. Jaeho Ryu, Naoki Hashimoto, and Makoto Sato. Influence of resolution degradation on distance estimation in virtual space displaying static and dynamic image. In *Proceedings of the International Conference on Cyberworlds (CW 05)*. IEEE, 2005.
31. David Drascic and Paul Milgram. Perceptual issues in augmented reality. In *Proceedings of the SPIE Stereoscopic Displays and Virtual Reality Systems III*, volume 2653, pages 123–134, San Jose, California, February 1996.
32. Thomas E. Whalen, Sylvie Noël, and John Stewart. Measuring the human side of virtual reality. In *Proceedings of the International Symposium on Virtual Environments, Human-Computer Interfaces and Measurement Systems (VECIMS 03)*, pages 8–12. IEEE, 2003.
33. ITU-T G.100/P.10 Amendment 2. Series g: Transmission systems and media, digital systems and networks international telephone connections and circuits – general definitions series p: Telephone transmission quality, telephone installations, local line networks. Technical report, International Telecommunication Union, 2008.
34. Lauralee Alben. Quality of experience: Defining the criteria for effective interaction design. *interactions*, 3(3):11–15, June 1996.

35. Ramesh Jain. Quality of experience. *Multimedia, IEEE*, 11(1):96–95, 2004.
36. Niamh McNamara and Jurek Kirakowski. Defining usability: Quality of use or quality of experience? In *Proceedings of the International Professional Communication Conference (IPCC 2005)*, pages 200–204. IEEE, 2005.
37. Xiaoxiang Lu, Ramon Orlando Morando, and Magda El Zarki. Understanding video quality and its use in feedback. In *Proceedings of the International Packet Video Workshop*, Pittsburgh, USA, 2002.
38. Tobias Sielhorst, Wu Sa, Ali Khamene, Frank Sauer, and Nassir Navab. Measurement of absolute latency for video see through augmented reality. In *Proceedings of the 6th International Symposium on Mixed and Augmented Reality (ISMAR 2007)*, pages 215–220. IEEE and ACM, 2007.
39. Adam L. Janin, David W. Mizell, and Thomas P. Caudell. Calibration of head-mounted displays for augmented reality applications. In *Proceedings of the Virtual Reality Annual International Symposium*, pages 246–255. IEEE, 1993.
40. Mihran Tuceryan and Nassir Navab. Single point active alignment method (SPAAM) for optical see-through HMD calibration for AR. In *Proceedings of the International Symposium on Augmented Reality (ISAR 2000)*, pages 149–158. IEEE and ACM, 2000.
41. Stephen R. Gulliver and Gheorghita Ghinea. The perceptual and attentive impact of delay and jitter in multimedia delivery. *IEEE Transactions on Broadcasting*, 53(2):449–458, June 2007.
42. ITU-T Recommendation BT.1082-1. Studies toward the unification of picture assessment methodology. Technical report, International Telecommunication Union, 1990.
43. ITU-T Recommendation BT.500-11. Methodology for the subjective assessment of the quality of television picture. Technical report, ITU-T, January 2002.
44. Yoo-Sok Saw. *Rate-quality optimized video coding*. Kluwer Academic Publishers, Norwell, MA, USA, 1999.
45. Zhenzhong Chen and King Ngi Ngan. Recent advances in rate control for video coding. *Signal Processing: Image Communications Magazine*, 22(1):19–38, January 2007.

46. Sun Yu and Ishfaq Ahmad. New rate control algorithm for MPEG-4 video coding. In *Proceedings of the Conference on Visual Communications and Image Processing*, volume 4671 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pages 698–709, January 2002.
47. Chi-Wah Wong, Oscar C. Au, and Hong-Kwai Lam. PID-based real-time rate control. In *Proceedings of the International Conference on Multimedia and Expo (ICME 04)*, volume 1, pages 221–224. IEEE, 2004.
48. Liquan Shen, Zhi Liu, Zhaoyang Zhang, and Xuli Shi. Rate control based on an incremental proportional-integral-differential algorithm. *Optical Engineering*, 46(7):077002–1–7, July 2007.
49. Arcangelo Bruna, Stewart Smith, Filippo Vella, and Filippo Naccari. JPEG rate control algorithm for multimedia. In *Proceedings of the International Symposium on Consumer Electronics*, pages 114–117. IEEE, 2004.
50. Dorgham Sisalem and Henning Schulzrinne. The loss-delay based adjustment algorithm: A TCP-friendly adaptation scheme. In *Proceedings of Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*, Cambridge, UK., 1998.
51. Manish Jain and Constantinos Dovrolis. End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput. *IEEE/ACM Transactions on Networking*, 11(4):537–549, August 2003.
52. Yegyalakshmi Easwaran and Miguel A. Labrador. Evaluation and application of available bandwidth estimation techniques to improve TCP performance. In *Proceedings of the 29th Annual International Conference on Local Computer Networks (LCN 04)*, pages 268–275. IEEE, 2004.
53. Mehmet U. Demircin and Peter van Beek. Bandwidth estimation and robust video streaming over 802.11e wireless LANs. In *Proceedings of the International Conference on Multimedia and Expo (ICME 2005)*, pages 1250–1253. IEEE, 2005.
54. Christophe Salzmann, Denis Gillet, and Philippe Müllhaupt. End-to-end adaptation scheme for ubiquitous remote experimentation. *Personal Ubiquitous Computing*, 13(3):181–196, March 2009.

55. Olivier Verscheure, Pascal Frossard, and Maher Hamdi. User-oriented QoS analysis in MPEG-2 video delivery. *Real-Time Imaging*, 5(5):305–314, October 1999.
56. G. Davini, D. Quaglia, J.C. De Martin, and C. Casetti. Perceptually-evaluated loss-delay controlled adaptive transmission of MPEG video over IP. In *Proceedings of the International Conference on Communications (ICC 03)*, volume 1, pages 577–581. IEEE, 2003.
57. Wenyu Jiang and Henning Schulzrinne. Modeling of packet loss and delay and their effect on real-time multimedia service quality. In *Proceedings of the International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV 2000)*, 2000.
58. Gregory K. Wallace. The JPEG still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1):xviii–xxxiv, 1992.
59. Zhihai He and Sanjit K. Mitra. A linear source model and a unified rate control algorithm for DCT video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(11):970–982, 2002.
60. Joong-Yun Lee and Hyun Wook Park. A rate control algorithm for DCT-based video coding using simple rate estimation and linear source model. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(9):1077–1085, September 2005.
61. Christian van der Branden Lambrecht. *Perceptual models and architectures for video coding applications*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne, 1996.
62. A.R. Prasad, R. Esmailzadeh, S. Winkler, T. Ihara, B. Rohani, B. Pinguet, and M. Capel. Perceptual quality measurement and control: Definition, application and performance. In *Proceedings of the 4th International Symposium on Wireless Personal Multimedia Communications*, volume 2, page 255, Aalborg, Denmark, 2001.
63. Chi-Wah Wong, O.C. Au, Bojun Meng, and Hong-Kwai Lam. Perceptual rate control for low-delay video communications. In *Proceedings of the International Conference on Multimedia and Expo (ICME 03)*, volume 3, pages III–361–III–364. IEEE, 2003.
64. Kazuo Takahata, Norihiko Uchida, and Yoshitaka Shibata. QoS control for real time video stream over hybrid network by wired and wireless LANs. In *Proceedings of the 17th International Conference*

- on *Advanced Information Networking and Applications (AINA 03)*, pages 45–51. IEEE, 2003.
65. Y. Bai and M.R. Ito. Network-level loss control schemes for streaming video. In *Proceedings of the International Conference on Multimedia and Expo (ICME 04)*, volume 1, pages 495–498. IEEE, 2004.
 66. Peter van Beek, Sachin Deshpande, Hao Pan, and Ibrahim Sezan. Adaptive streaming of high-quality video over wireless LANs. In *Proceedings of the Conference on Visual Communications and Image Processing*, volume 5308 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pages 647–660. SPIE, January 2004.
 67. Peter van Beek and Mehmet Umut Demircin. Delay-constrained rate adaptation for robust video transmission over home networks. In *Proceedings of the International Conference on Image Processing (ICIP 2005)*, volume 2, pages 173–176. IEEE, 2005.
 68. Nicola Cranley, Liam Murphy, and Philip Perry. Perceptual quality adaptation (PQA) algorithm for 3GP and multi-tracker MPEG-4 content over wireless IP networks. In *Proceedings of the 5th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 04)*, volume 3, pages 2107–2112. IEEE, 2004.
 69. Nicola Cranley, Philip Perry, and Liam Murphy. Dynamic content-based adaptation of streamed multimedia. *Journal of Network and Computer Applications*, 30(3):983–1006, August 2007.
 70. Damien Perritaz, Christophe Salzmänn, Denis Gillet, Olivier Naef, Jacques Bapst, Barras Frédéric, Elena Mugellini, and Omar Abou Khaled. *Human Machine Interaction*, volume LNCS 5440 of *State-of-the-Art Surveys*, chapter 6th Sense - Toward a Generic Framework for End-to-End Adaptive Wearable Augmented Reality, pages 283 – 315. Springer-Verlag, Berlin/Heidelberg, Lalanne Denis and Kohlas Jürg edition, March 2009.
 71. Carl Runge. Über empirische funktionen und die interpolation zwischen äquidistanten ordinaten. *Zeitschrift für Mathematik und Physik*, 46:224–243, 1901.
 72. H. B. Curry and Schoenberg Isaac Jacob. On pólya frequency functions iv: the fundamental spline functions and their limits. *Journal d'analyse mathématique*, 17(1):71–107, 1966.

73. Carl de Boor. *A practical guide to splines*, volume 27. Springer-Verlag, 1978.
74. Charles K. Chui. *Multivariate Splines*. Society for Industrial Mathematics, 1988.
75. Yue Wang and Jeremy Taylor. Monotone constrained tensor-product B-spline with application to screening studies. Technical Report 1022, University of Michigan, 2004.

Index

- ρ -domaine, 81, 82
- adaptation, 105, 124
 - contextuelle, 109, 121
 - de transmission, 50, 109
- affine, 82
- algorithme direct, 85, 91
- approche système, 6–8, 48
- architecture, 108
- B-spline, 140
- caméra virtuelle, 24
- coefficient, 27, 38, 138, 140, 151
- contenu, 43, 81
- contexte, 5, 18
- contrôle de débit, 4, 7, 43, 48, 95
- contrôleur, 49
- coupe, 119, 145
- DCT, 62, 67
- débit, 53
 - de référence, 50
 - disponible, 50
 - du flux vidéo, 56, 72
- délai
 - de transmission, 52
 - global, 13, 20, 23
- encodage vidéo, 44, 62, 135
- entrée unique, 56, 61
- erreur d’alignement, 3, 13, 21
- espace
 - d’encodage, 19
 - d’encodage augmenté, 19, 111
 - linéaire, 74
- espace de couleurs, 64
- évaluation subjective, 33
- expérience utilisateur, 5, 9, 10, 109, 134
- fonction réalisable, 111
 - augmentée G, 113
 - g, 56, 61, 78
 - h, 72, 74
- fréquence, 20, 21

- histogramme, 87
- hypersurface, 19, 25, 34
- intersection, 116
- mapping, 8, 61, 75
 - Φ , 75
 - Φ_1 , 76
 - Φ_2 , 76, 80, 85, 91
- mesures expérimentales, 35, 92, 124
- métrique
 - contenu, 56, 98
 - qualité, 12
 - qualité d'expérience, 16
 - qualité de service, 50, 109
- modèle
 - de l'encodeur, 57
 - contextuel de qualité d'expérience, 20, 28
 - de transmission, 51, 52
 - générique de qualité d'expérience, 25, 26
 - identifié de qualité d'expérience, 32, 39
 - qualité d'expérience, 18, 19
- monotonie, 23, 24, 26, 38, 137, 148
- noeud, 27, 38, 138, 140, 151
- optimisation, 35, 116
- paramètres d'encodage, 18, 21, 44
- perception, 1, 11, 13
- quadrants, 76, 111
- qualité d'expérience, 7, 9, 16
- quantification, 20, 21, 68, 87
- réalité augmentée, 2, 134
- rétroaction, 6, 8, 49, 109
- simulation, 28, 121
- spline, 137, 140
 - monovariante, 138
 - multivariante, 143
- streaming, 12, 45, 107
- système, 48
 - transmission, 50
 - AR, 109, 120
 - encodeur, 50, 53
 - transmission, 51
- variables
 - contextuelles, 18, 23, 24
 - de conception, 18
 - du modèle, 20
- vitesse de la tête, 13–15, 20, 24

Acronymes

YCbCr luminance-chrominance.

pose combinaison de position et orientation.

AR *Augmented Reality* pour réalité augmentée.

BIBO *Bounded Input Bounded Output* pour entrée bornée sortie bornée.

CODEC *COder-DECoder* pour encodeur-décodeur.

DCT *Discrete Cosine Transform* pour transformée en cosinus discrète.

FIFO *First In First Out* pour premier entré premier sorti.

HMD *Head Mounted Display* pour visiocasque.

HVS *Human Visual System* pour système visuel humain.

ITU International Telecommunication Union.

JPEG Joint Photographic Experts Group.

LDA *Loss-Delay Adjustment* pour ajustement perte-délai.

M-JPEG Motion Joint Photographic Experts Group.

MPEG Moving Picture Experts Group.

OAT *Optimal Adaptation Trajectory* pour trajectoire d'adaptation optimale.

PDM *Perceptual Distortion Metric* pour métrique de distorsion perceptuelle.

PID Proportionnel Intégral Dérivé.

PQA *Perceptual Quality Adaptation* pour adaptation perceptuelle de qualité.

PSNR *Peak Signal to Noise Ratio* pour rapport crête signal sur bruit.

QOE *Quality of Experience* pour qualité d'expérience.

QoP *Quality of Perception* pour qualité de perception.

QoS *Quality of Service* pour qualité de service.

QoU *Quality of Use* pour qualité d'utilisation.

RGB Red Green Blue.

SISO *Single Input Single Output* pour à une entrée et une sortie.

TMN8 *video codec* Test Model, Near-term, version 8.

VM8 Verification Model, version 8.

VR *Virtual Reality* pour réalité virtuelle.

Symboles

B B-spline.

C^k ensemble continuellement dérivable.

D profondeur de couleurs.

F matrices des coefficients DCT.

G fonction réalisable augmentée dans l'espace d'encodage augmenté.

H hauteur de l'image.

M modèle de QoE.

M_c modèle de QoE contextuel.

M_g modèle générique de QoE.

M_i modèle de QoE identifié.

N nombre de blocs d'une image.

N_0 nombre de coefficients nuls.

N_T nombre total de coefficients.

$N_{\bar{0}}$ nombre de coefficients non-nuls.

$P(x)$ polynôme.

$R(\rho)$ modèle linéaire dans le ρ -domaine.

S fonction spline.

UX expérience utilisateur subjective.

V volume de données dans le tampon.

W largeur de l'image.

Φ mapping entre l'espace linéaire et l'espace d'encodage.

Φ_1 mapping entre la période p et la fréquence f .

Φ_2 mapping entre la variable linéaire l et la quantification q .

α coefficients d'une spline.

δ décalage de la droite affine dans l'espace linéaire.

\mathbb{R} ensemble des nombres réels.

Q matrices des coefficients quantifiés.

S taille de l'image.

\mathcal{T} table de quantification.

\mathcal{W} facteur de pondération.

a vecteur des valeurs absolues des coefficients DCT.

d résolution des composantes de luminance et chrominance.

f matrice des pixels.

h histogramme des occurrences.

q vecteur de quantification.

r seuil de quantification.

s facteur d'échantillonnage.

ρ pourcentage de coefficients nuls de la DCT.

τ noeuds d'une spline.

θ contenu dans le ρ -domaine.

a gain du contrôleur.

a_{lim} gain limite du contrôleur.

c métrique de contenu du flux vidéo.

- d délai global.
 d_t délai de transmission.
 e erreur.
 f fréquence de génération des images.
 g fonction réalisable dans l'espace d'encodage.
 h fonction réalisable dans l'espace linéaire.
 j fonction d'intersection entre le modèle contextuel et la fonction réalisable augmentée.
 k variable de temps discrète.
 l variable linéaire liée à la quantification.
 m_c mesures contextuelles.
 n métriques liées à la transmission.
 o taille de l'image compressée.
 p période de temps entre deux images successives.
 q facteur de quantification.
 r débit (quantité par unité de temps).
 r_a débit disponible.
 r_r débit de référence.
 s vitesse de rotation de la tête.
 t variable de temps continue.
 u entrée unique du modèle de l'encodeur.
 v_c variables contextuelles.
 v_e paramètres d'encodage.
 w ordre d'une fonction.
 x entrée (variable ou matrice).
 y sortie (variable ou vecteur).
- indice se référant à toutes les valeurs.
 - ' élément lié à l'élément correspondant.

Termes anglophones

buffer tampon.

end-to-end global.

header en-tête.

jitter gigue.

mapping mise en correspondance.

multi-pass plusieurs passes.

pinhole camera model modèle simple de caméra.

quantization quantification.

rate control contrôle de débit.

rate débit.

reference-based basées sur une référence.

registration alignement.

single-pass une passe.

smartphone téléphone intelligent.

streaming diffusion de flux continu.

thin client client léger.

tracking positionnement.

Damien Perritaz

Marchet 52
1740 Neyruz (FR)

damien.these@perritaz.ch
+41-79/200-24-90

28 ans – 8 juin 1982
Nationalité suisse

Formation

2006-2010 : Docteur ès Sciences au Laboratoire d'Automatique,
École Polytechnique Fédérale de Lausanne
2001-2006 : Master ès Sciences en Microtechnique, École Polytech-
nique Fédérale de Lausanne

Publications

1. Damien Perritaz, Christophe Salzmann, Denis Gillet, Olivier Naef, Jacques Bapst, Barras Frédéric, Elena Mugellini, and Omar Abou Khaled. *Human Machine Interaction*, volume LNCS 5440 of *State-of-the-Art Surveys*, chapter 6th Sense - Toward a Generic Framework for End-to-End Adaptive Wearable Augmented Reality, pages 283 – 315. Springer-Verlag, Berlin/Heidelberg, Lalanne Denis and Kohlas Jürg edition, March 2009.
2. Damien Perritaz, Christophe Salzmann, and Denis Gillet. Quality of experience adaptation in augmented reality. In *Proceedings of the International Conference on Systems, Man, and Cybernetics*, San Antonio, USA, October 11-14 2009. IEEE.

3. Davide Bucciari, Damien Perritaz, Philippe Mullhaupt, Zhong-Ping Jiang, and Dominique Bonvin. Velocity-scheduling control for a unicycle mobile robot: Theory and experiments. *IEEE Transactions on Robotics*, 25(2):451–458, 2009.
4. Damien Perritaz, Christophe Salzmann, and Denis Gillet. User perception model for wearable supervision systems. In *Proceedings of the 4th International Conference on Mobile Technology, Applications and Systems and the 1st International Symposium on Computer Human Interaction in Mobile Technology (Mobility 2007)*, Singapore, September 2007. ACM.
5. Davide Bucciari, Damien Perritaz, Philippe Mullhaupt, Zhong-Ping Jiang, and Dominique Bonvin. Experimental results for a nonholonomic mobile robot controller enforcing linear equivalence asymptotically. In *Conference on Industrial Electronics and Applications*, pages 1–6. IEEE, 2006.
6. Denis Gillet, Christophe Salzmann, Yvan Michellod, and Damien Perritaz. Hands-on introduction to automatic control with focus on teleoperation and real-time interaction. In *Proceedings of the Symposium on Advances in Control Education (ACE 2006)*, 2006.