Clonage de visage et spatialisation vidéo: outils pour la téléconférence virtuelle

Face Cloning and Video Spatialization: Tools for Virtual Teleconference

par J.-L. Dugelay[1], K. Fintzel[1/3], S. Valente[1] & H. Delingette[2]

[1] *Institut EURECOM, Département Communications Multimédia*
2229, route des Crêtes, B.P. 193, F-06904 Sophia–Antipolis Cedex, France
traivi@eurecom.fr, *tél: 04 93 00 26 33 fax: 04 93 00 26 27*

[2] *INRIA, Projet Epidaure*
B.P. 93, F-06902 Sophia–Antipolis Cedex, France

[3] *Espri Concept*
Les Taissounières HB2 B.P. 277, F-06905 Sophia–Antipolis Cedex, France

résumé

Dans cet article, nous proposons des algorithmes de traitement d'images vidéo (tels que le clonage de visages et la spatialisation vidéo) qui peuvent être utilisés pour définir de nouveaux systèmes de vidéoconférence offrant plus de "confort d'utilisation" que les systèmes actuels, malgré des liaisons très bas-débit. Ce nouveau concept repose sur la métaphore d'une salle de réunion virtuelle où les utilisateurs pourront choisir leur place.
En particulier, nous proposons des modules de clonage vidéo pour représenter les participants par l'intermédiaire de modèles synthétiques 3D de leur visage, obtenus par création de maillages simplexes sur des données Cyberware. Ces modèles sont visualisables sous des points de vue différents de celui de la caméra qui analyse les mouvements des participants.
Par ailleurs, le réalisme de l'espace de réunion virtuelle est renforcé par des techniques de spatialisation vidéo qui a pour but de créer des points de vue inédits à partir d'images statiques non-calibrées d'une salle de réunion existante.
mots clés: téléconférence virtuelle, réseaux très bas-débit, modélisation et traitements d'images, clonage de visages, spatialisation vidéo.

abstract

In this paper, we propose powerful virtual image processing tools (face cloning and video spatialization) which can be useful to design new teleconferencing systems offering a better comfort for users even if very low bit rate links are used. These tools allow a new teleconferencing concept, relying on the metaphor of a virtual meeting room where the participants can choose their position and point of view.
In particular, we propose video cloning modules to represent the participants via $3D$ synthetic models of their face, contructed from range data with simplex meshes. These models are meant to be visualized under a point of view different from the camera which analyses the facial motion of the speakers.
Besides, the realism of the virtual meeting room is improved by video spatialization techniques,

which aims at synthesizing new points of view from a limited set of uncalibrated views of an existing room.
Keywords: virtual teleconferencing, low bit rate networks, $3D$ modeling, video processing, video cloning, video spatialization.

1 Introduction

Il existe aujourd'hui deux tendances concernant les systèmes de téléconférence couramment utilisés:

- les standards (norme H320 pour la définition des systèmes de vidéoconférence [1]) utilisant le réseau RNIS (Réseau Numérique à Intégration de Services) de couverture très large qui garantit une bande passante ainsi que des paramètres de "Qualité de Service" satisfaisants.
- les autres (par exemple les outils MBone [2] vic et vat [3]) supportés par le réseau INTERNET, qui n'a cessé de croître ces dernières années constituant en particulier un excellent support pour les communications multipoints [4].

Cependant, aucune des solutions mises en oeuvre jusqu'à maintenant ne s'est révélée pleinement satisfaisante ; chacune d'entre-elles présente des limites tant d'un point de vue technique que d'un point de vue ergonomique. En effet, d'un point de vue technique les systèmes de vidéoconférence RNIS actuels sont très limités dans le cas de communications multipoints, alors que les outils INTERNET qui gèrent correctement ce type de multi-communications n'assurent pas, de par la surcharge du réseau, une Qualité de Service et une capacité de bande passante suffisantes.
Plus significatif encore, d'un point de vue ergonomique aucune de ces deux solutions ne produit pour l'utilisateur, un réel sentiment de téléprésence puisqu'elles ne lui offrent qu'une vue 2D des autres participants sans cohérence de position relative et sans environnement commun.
Même si des travaux plus récents en analyse d'images orientée objet [5, 6] ou dédiés "tête et épaules" essaient de pallier aux limites techniques précédentes, aucun de ces nouveaux développements ne vise une amélioration des limites ergonomiques des systèmes de visioconférence : c'est là que se situe l'originalité de notre travail. En effet, le concept de téléconférence virtuelle que nous proposons opère une rupture complète par rapport aux travaux précédents, puisqu'il a pour but d'immerger tous les participants à une visioconférence dans un même espace de réunion virtuelle, afin de leur offrir ce sentiment de présence qui fait jusqu'ici cruellement défaut aux systèmes de visioconférence. Ce travail nécessite l'utilisation de techniques de modélisation et de traitement d'image avancées telles que le clonage vidéo permettant de gérer les mouvements globaux et locaux de chacun des participants, et la spatialisation vidéo pour contrôler la cohérence des images de fond de la scène. Ces travaux doivent bien sûr nous permettre de restituer les mouvements de chacun des participants de façon pertinente via des liaisons très bas-débit, mais nous permettent également de restituer des fonctionnalités existant en réunions "réelles" ou d'en envisager de nouvelles : chaque interlocuteur ayant par exemple la possibilité de choisir la place qu'il occupera pendant la réunion (le point de vue n'étant plus unique et imposé), le type d'espace dans lequel se déroulera cette réunion et même le modèle synthétique qui le représentera.
Ces nouveaux outils de traitement d'image pour les télécommunications, clonage de visages et spatialisation vidéo sont détaillés respectivement dans les sections 2 et 3. La dernière section, plus prospective, est consacrée aux nouvelles perspectives du projet ainsi qu'à l'étude des possibilités d'intégration de ce type de système de téléconférence virtuelle aux stations de travail, réseaux et interfaces standards tels que les PC, Internet et VRML.

2 Clonage vidéo

Dans le contexte de notre projet de télé-virtualité, l'intérêt du clonage vidéo est le suivant :

- fournir aux utilisateurs une représentation 3D réaliste des autres participants, qui peut être visualisée sous n'importe quel angle de vue, suivant la position initiale que chaque personne veut ou est sensée occuper dans la salle de réunion virtuelle ;
- éviter de transmettre des images via le réseau et n'envoyer qu'une représentation compacte sous forme de paramètres permettant l'animation du modèle et nécessitant une bande passante aussi faible que possible.

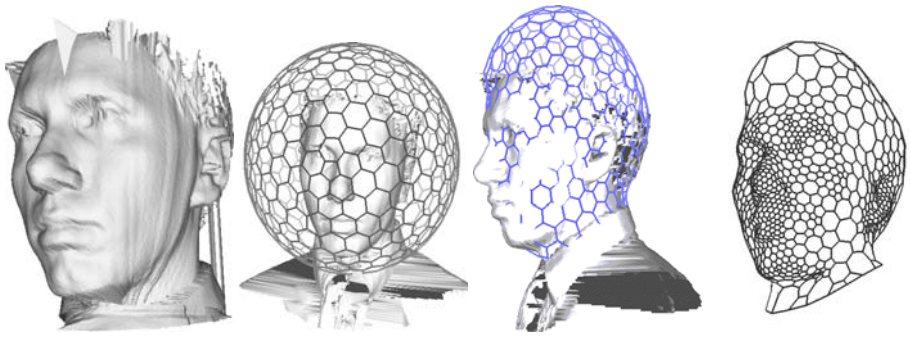
2.1 Travaux précédents

Des algorithmes de clonage vidéo sont couramment utilisés pour l'animation globale d'un modèle (correspondant à la position et à l'orientation de l'intervenant dans l'espace 3D), ainsi que pour son animation locale (révélant ses expressions faciales courantes). Dans la littérature, on trouve plusieurs références concernant le clonage vidéo, comme [7, 8, 9, 10]. La plupart d'entre-elles considèrent que l'intervenant regarde la caméra et restreignent les mouvements globaux du visage pour éviter des grandes rotations où les centres d'intérêt des visages deviennent difficiles à suivre. D'autre part, on peut déplorer le manque de réalisme des modèles faciaux utilisés dans [7, 9, 10] : produits "à la main", ils représentent souvent le sujet d'une manière très stylisée. Ces modèles artificiels, ou "avatars", sont néanmoins largement utilisés car du fait de leur géométrie simplifiée, leur animation et leur manipulation sont aisées. En ce qui concerne la création de clones hautement réalistes, on peut citer les travaux de modélisation anatomique et physique de Terzopoulos et Waters qui retravaillent des modèles Cyberware [11] pour les rendre manipulables par un système générique de plus haut niveau [8]. Toutefois, leurs modèles restent trop complexes pour être animés en temps–réel, et leur algorithme d'analyse, basé sur des contours actifs, nécessite que des contours noirs soient marqués sur le visage de l'utilisateur.
L'intégration du clonage vidéo dans un système de téléconférence impose des contraintes très spécifiques sur les modules d'analyse et de restitution des participants : un tel système doit pouvoir opérer en temps–réel, dans un environnement avec un éclairage sans contrainte particulière, et sans maquillage sur l'utilisateur. De plus, celui–ci ne doit pas être restreint dans ses mouvements, et son clone doit être le plus réaliste possible. Dans cet article, nous décrivons des algorithmes permettant de résoudre le problème du suivi et de la détermination des position et orientation de l'utilisateur sans contrainte (d'environnement, d'éclairage ou de maquillage), avec un modèle texturé réaliste. Dans la section 2.2, nous décrivons la construction du modèle $3D$. En section 2.3, nous présentons une boucle analyse/synthèse s'appuyant sur le réalisme du modèle pour le suivi du visage. La section 2.4 propose une adaptation photométrique du modèle à l'éclairage de l'utilisateur. Ensuite, nous reformulons en section 2.5 un algorithme d'appariement de blocs synthétiques et réels. Enfin, en section 2.6, nous discutons des résultats expérimentaux obtenus.

2.2 Modélisation de Visage

Comme point de départ, nous utilisons actuellement des données Cyberware [11] pour construire le modèle du visage d'une personne. Les données brutes consistent en deux fichiers : une carte de profondeur en coordonnées cylindriques de la tête de la personne, et une texture, également cylindrique, à plaquer sur la forme $3D$ du visage. Cependant, ces données ne peuvent être utilisées directement, car elles sont trop denses (en moyenne $1,4$ millions de primitives) et il y a souvent des points isolés (comme sur la figure 1(a)), ou des zones manquantes (typiquement le haut du crâne).

Pour obtenir un modèle à la fois réaliste et manipulable en temps–réel, nous devons retravailler les données brutes afin d'en diminuer le nombre de primitives, mais en gardant un niveau de détails suffisant autour des points caractéristiques du visage comme les lèvres ou les yeux. Nous avons développé un système de construction de maillages simplexes [12] dans ce but. A l'inverse des approches classiques, les maillages simplexes sont traités comme des maillages discrets, sans aucune paramétrisation, et peuvent être directement convertis en maillages triangulaires pour une exploitation ultérieure plus simple.
Dans la figure 1, nous montrons les différentes étapes de la construction du maillage sur des données Cyberware présentant des points isolés et des trous au niveau du haut de la tête. Le modèle simplexe est d'abord initialisé en tant que sphère, puis déformé pour coller grossièrement au visage. La dernière étape consiste à raffiner le maillage simplexe sur des zones précises, où des primitives sont automatiquement ajoutées suivant la distance entre les données du visage et la courbure du maillage (figure 1(d)).



(a) (b) (c) (d)

FIG. 1 – *Construction d'un modèle géométrique à partir de données Cyberware : (a) données initiales ; (b) initialisation sphérique ; (c) déformation principale ; (d) raffinement du maillage — nous avons sélectionné interactivement les zones d'intérêt du visage (comme le menton, les oreilles, les yeux, le nez...) où le maillage doit offrir plus de précision. Le maillage final est composé de 2048 primitives, et la construction a été réalisée en moins de 5 minutes sur une DEC Alphastation 233 Mhz.*

Pour finir, on associe à chaque primitive $3D$ du maillage les coordonnées (u, v) de texture du point le plus proche trouvé dans les données initiales. Si la primitive correspond à un trou des données brutes (comme pour les cheveux), on obtient les coordonnées de texture par projection de la primitive sur le cylindre de la texture.
Au final, notre algorithme produit donc un modèle géométrique texturé, réaliste et de faible complexité, compatible avec des manipulations temps–réel.

2.3 Boucle d'Analyse/Synthèse

Grâce au réalisme des modèles de visages, nous avons implanté un algorithme de suivi des mouvements globaux des participants (translations et rotations de leur tête devant la caméra qui les filme), par analyse des images filmées par une simple caméra de station de travail, qui recherche les éléments caractéristiques de leur visage dans l'image $2D$ tels qu'ils sont définis par la synthèse de leur modèle.
La boucle de suivi opère de la manière suivante (figure 2) :

– un filtre de Kalman prédit les position et orientation $3D$ du visage à l'instant t en prenant en compte les observations des positions $2D$ des points caractéristiques dans toutes les images jusqu'à $t - 1$;

– à partir des paramètres $3D$ estimés et du modèle de l'interlocuteur, les éléments caractéristiques du visage sont synthétisés, en prenant en compte l'échelle et les déformations géométriques des centres d'intérêt du visage dues à la position du locuteur devant la caméra, ainsi que l'éventuelle apparition du fond de la scène dans le voisinage des centres d'intérêt. De plus, grâce au module de compensation photométrique $3D$ décrit en section 2.4, les motifs générés se rapprocheront de l'illumination du visage (qui varie naturellement suivant la position et l'orientation du locuteur dans l'espace). La taille des imagettes synthétisées dépend de la position prédite du modèle dans l'espace, en variant typiquement autour de 20×20 pixels ;

– un algorithme d'appariement de blocs modifié recherche les régions du visage synthétisées dans l'image prise à l'instant t ;

– le filtre de Kalman incorpore les positions $2D$ des régions faciales appariées dans le plan image, et produit une nouvelle estimation de la position et de l'orientation en $3D$ du visage pour l'instant $t+1$.

Le filtre de Kalman a plusieurs rôles dans ce schéma : en premier lieu, il vise à prédire les positions initiales $2D$ pour la procédure d'appariement de blocs ; il estime la pose $3D$ du locuteur à partir des observations $2D$ dans le plan image ; et enfin, il contrôle la synthèse du modèle, en prenant soin de générer les bonnes position et échelle du modèle pour qu'il puisse s'aligner avec la vue réelle du locuteur malgré la caméra non–calibrée. Ceci est dû au fait que le modèle d'observation du filtre est dérivé, non d'un modèle classique de caméra, mais de la projection perspective opérée par la procédure de synthèse. Quant au modèle dynamique qui décrit l'évolution du système, il s'agit d'une équation de mouvement à accélération constante. Le vecteur d'état du filtre est donc composé des position et orientation du modèle $3D$ dans l'espace synthétique, et de leurs dérivées premières et secondes.
Notre coopération Analyse/Synthèse optimisée, résultant du réalisme des modèles faciaux et du contrôle par filtrage de Kalman, permet un suivi de visage plus robuste sans marqueur ni maquillage, et autorise de grandes rotations hors du plan image (comme sur la figure 2).

2.4 Compensation d'illumination

Les figures 3(a) et 3(b) mettent en évidence le fait que si l'on utilise une illumination par défaut pour le modèle synthétique, il y aura trop de différences photométriques entre les motifs faciaux générés et l'image du locuteur prise dans un environnement réel pour qu'un algorithme d'appariement de blocs (basé sur la luminance) puisse fonctionner correctement. Dans le contexte de notre application, les différences d'illumination sont dues à la fois au locuteur qui ne porte pas nécessairement de maquillage pour éviter les réflexions lumineuses sur son visage, et à son environnement qui a un éclairage quelconque.
Pour résoudre ce problème, nous opérons une compensation d'illumination lors de la synthèse du modèle en utilisant les possibilités d'éclairage de la librairie OpenGL, avec des sources lumineuses ambiante, diffuse et spéculaire (figure 3 (c)) à des positions fixes, mais ayant une intensité variable. L'avantage de cette technique est que la compensation d'illumination est prise en charge directement par le module de synthèse d'image en $3D$ (éventuellement en tirant partie d'accélérations matérielles disponibles sur certaines cartes graphiques) à l'inverse d'autres techniques qui agissent en $2D$ sur l'image lors de post–traitemements souvent coûteux en temps de calcul. Les intensités des sources lumineuses sont calculées suivant un critère quadratique au début de la session d'analyse, en comparant une vue réelle de l'utilisateur avec une vue de son modèle aligné (figure 3).
Il est évident qu'une telle modélisation synthétique ne peut compenser exactement l'illumination réelle du locuteur dans n'importe quelle position, mais elle permet tout de même de minimiser les

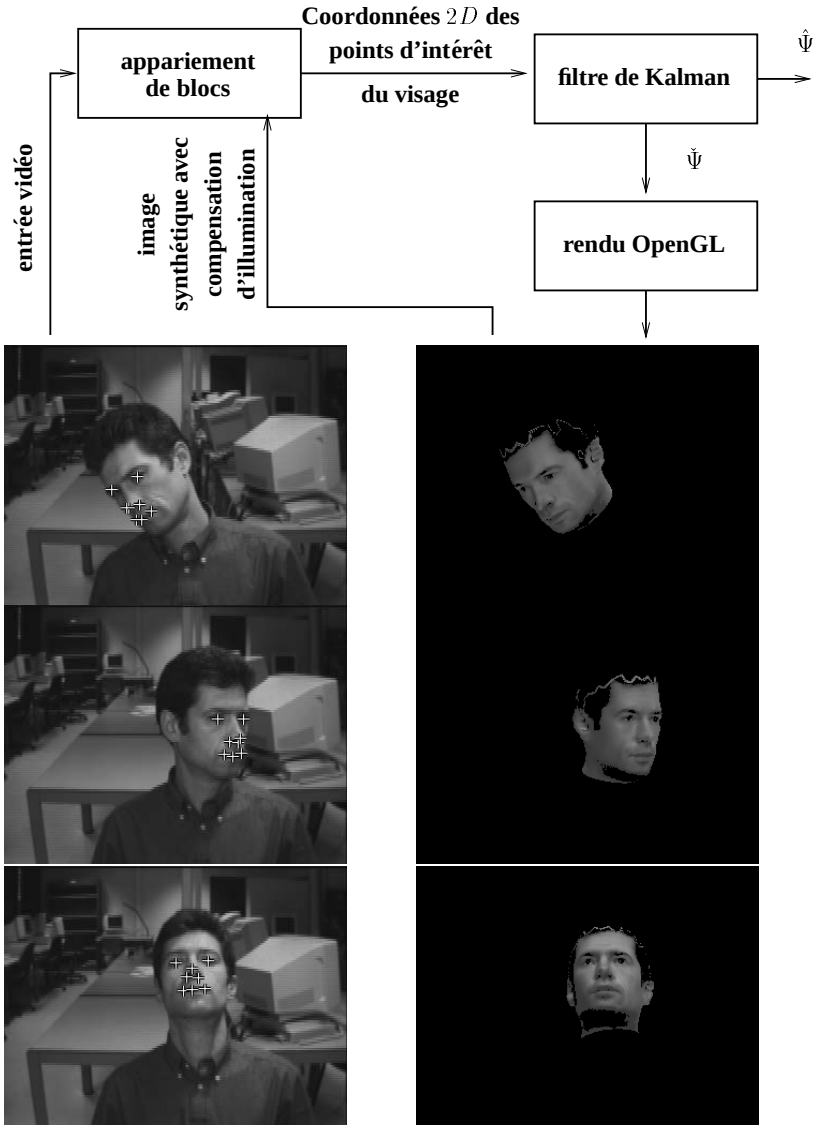


FIG. 2 – *Boucle d'Analyse/Synthèse fondée sur un filtre de Kalman et un retour synthétique — $\check{\Psi}$ et $\hat{\Psi}$ sont les positions et orientations $3D$ du visage suivi, respectivement prédites (estimation* a priori*) et filtrées (estimation* a posteriori*). Les exemples montrés sont extraits d'une séquence vidéo de 30 secondes capturée avec une résolution de 320×242 pixels, à 10 images par seconde.*

différences entre les images analysées et synthétisées pour qu'un appariement des blocs soit possible. Nous invitons les personnes désirant obtenir plus de détails théoriques et expérimentaux à se procurer la référence [13].

2.5 Appariement de Blocs avec des Motifs Synthétiques

Nous avons vu précédemment que le recours à un visage synthétique permettait de "prédire" les variations géométriques et d'illumination des points caractéristiques de l'utilisateur au cours de ses mouvements. Par ailleurs, puisque le modèle est synthétisé sur un fond noir uniforme, les blocs fournis par le module de synthèse peuvent clairement indiquer à l'analyse les zones où l'environnement du locuteur risque de se confondre avec les points suivis sur son visage. Toutefois, un algorithme d'appariement de blocs classique a toutes les chances d'être mis en difficulté pour deux raisons : la première est qu'il peut subsister des différences photométriques entre les blocs synthétiques et réels, et la seconde est que le fond noir, s'il apparaît dans les blocs

(a) (b) (c)

FIG. 3 – *Compensation d'illumination sur un visage réel — de gauche à droite : le modèle facial avec une illumination par défaut, le locuteur dans un environnement réel, et le même modèle avec des lumières synthétiques.*

synthétiques, va intervenir dans le score d'appariement, et conduire à des déviations.

2.5.1 Reformulation de l'Appariement de Blocs

Prolongeant la théorie présentée dans [14], nous proposons d'adapter la formulation classique de l'appariement différentiel de blocs (parfois appelée *pattern correlation* dans la littérature) pour récupérer les faiblesses du modèle photométrique en incluant un facteur d'échelle et un terme additif aux pixels synthétiques.
L'algorithme différentiel est obtenu en considérant qu'un bloc de référence à l'instant $t = 0$, noté $\boldsymbol{I}(0,0)$ (tous les pixels sont placés dans un vecteur) peut subir des petites perturbations $\boldsymbol{\mu} = (\mu_1, \cdots, \mu_n)^T$ (le plus souvent des déplacements en X et Y). Le développement limité du bloc entre deux images consécutives s'écrit :

$$\boldsymbol{I}(\boldsymbol{\mu}, \tau) = \boldsymbol{I}(0,0) + \boldsymbol{M}\boldsymbol{\mu} + \boldsymbol{I}_t\tau + \text{termes d'ordre supérieur} \quad (1)$$

avec $\boldsymbol{M} = [\frac{\partial \boldsymbol{I}}{\partial \mu_1}(0,0)|\cdots|\frac{\partial \boldsymbol{I}}{\partial \mu_n}(0,0)]$ and $\boldsymbol{I}_t = \frac{\partial \boldsymbol{I}}{\partial t}(0,0)$. Les perturbations $\boldsymbol{\mu}$ sont données, au sens des moindres carrés, par :

$$\boldsymbol{\mu} = -(\boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T\boldsymbol{I}_t \quad (2)$$

Dans l'équation 1, les perturbations $\boldsymbol{\mu}$ sont suffisamment générales pour représenter une rotation dans le plan image ou un zoom du bloc, et nous y ajoutons un facteur d'échelle et un offset sur les valeurs des pixels ($\frac{\partial \boldsymbol{I}}{\partial \text{facteur de luminance}}(0,0) = \boldsymbol{I}(0,0)$ et $\frac{\partial \boldsymbol{I}}{\partial \text{lum. offset}}(0,0) = (1,\cdots,1)^T$).
Remarquons que cette formulation autorise autant de degrés de liberté sur l'appariement que l'on désire, sans pour autant alourdir les calculs dans la détermination itérative de $\boldsymbol{\mu}$: en effet, le filtre de Kalman n'a besoin que du déplacement $2D$ X et Y du bloc pour retrouver la position et l'orientation de la tête suivie, si bien qu'une fois la matrice $(\boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T$ déterminée, seuls les paramètres de translation du bloc (μ_1, μ_2) doivent être calculés, et seules les deux premières lignes de $(\boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T$ sont utilisées, alors que d'autres degrés de liberté ont été inclus dans la formulation (comme des rotations $2D$ ou l'offset sur les pixels par exemple).

2.5.2 Prise en Compte de l'Arrière–Plan lors de l'Appariement

La prise en compte des pixels appartenant au fond noir derrière le modèle synthétique renforce la boucle de suivi lors des grandes rotations de l'utilisateur hors du plan image (comme sur la figure 4) : dans ce cas, les régions caractéristiques de son visage deviennent trop proches de l'arrière–plan pour être appariées sans ambiguïté avec des motifs synthétiques n'ayant pas le même fond en utilisant un algorithme classique.

FIG. 4 – *L'appariement de blocs, tel que nous l'avons reformulé, n'est pas perturbé par l'arrière–plan de la scène.*

Pour que les blocs synthétiques soient plus sélectifs (ou autrement dit que la partie "utile" des blocs ne soit pas nécessairement rectangulaire), leurs pixels sont classés en deux sous–ensembles, $\boldsymbol{I}|_F$ et $\boldsymbol{I}|_B$, suivant qu'ils correspondent au visage ou à l'arrière–plan. Si l'équation 2 est interprétée comme la corrélation de la différence entre le bloc synthétique et le bloc réel $\boldsymbol{I}_t$ avec la matrice $-(\boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T$, alors $\boldsymbol{I}_t|_F$ (la différence restreinte au sous–ensemble $\boldsymbol{I}|_F$) est la contribution des pixels du visage aux perturbations $\boldsymbol{\mu}$.
En pratique, quand un pixel d'arrière–plan est détecté dans un bloc synthétique, sa différence avec son correspondant réel est mise à zéro dans $\boldsymbol{I}_t$. Ainsi, l'arrière–plan ne perturbe pas le résultat numérique de la corrélation, et l'algorithme peut trouver le bon appariement dans l'image réelle.

2.6 Discussion sur la Robustesse du Suivi

La résultat de notre algorithme de suivi global peut être vu dans une séquence Mpeg disponible sur le WWW [15]. Sa vitesse sur une station de travail dépend principalement des possibilités d'accélération graphique et de la vitesse d'acquisition vidéo. Sur une station graphique d'entrée de gamme ($O2$ SGI), le taux d'analyse avec 12 régions suivies est :

- 1 image par seconde, lorsqu'on génère des blocs de référence synthétiques et met à jour le filtre de Kalman à chaque image ;
- 10 images par seconde, quand les blocs de référence ne sont pas resynthétisés à chaque image, mais en conservant le filtre de Kalman (dans ce cas, le système peut devenir sensible aux très grandes rotations) ;
- au rythme de l'acquisition vidéo, quand ni les blocs de référence, ni le filtre de Kalman ne sont mis à jour (le système suit simplement des régions en $2D$, sans estimer la position et l'orientation $3D$ du locuteur, et devient très sensible aux rotations).

En fait, l'appariement de bloc en tant que tel donne de bons résultats, malgré les différences entre réel et synthétique, même lors des très grandes rotations où l'arrière–plan pourrait poser problème (comme sur la figure 4). D'après notre expérience, la principale difficulté pour obtenir un système robuste est le réglage des paramètres du filtre de Kalman : il faut en effet fixer les variances des bruits des modèles d'observation et dynamique du filtre. D'un côté, si les bruits sont trop faibles, le filtre devient très instable. De l'autre, si les bruits sont trop forts, le filtre ne tient plus compte des observations données par l'appariement, et souffre d'une trop grande inertie qui l'empêche de suivre l'utilisateur lorsqu'il change de direction. La difficulté de ce réglage est propre à tous les problèmes d'estimation par filtrage de Kalman, mais dans notre exemple, nous ne disposons pas vraiment de modèles physiques pour déterminer des niveaux de bruit adéquats de manière systématique pour l'observation des appariements et pour le modèle dynamique à accéleration constante, car les mouvements $3D$ de la personne oscillent lentement autour de la

position initiale. Un modèle dynamique entrainé serait certainement plus approprié [16]. Le réglage du filtre de Kalman actuel reste donc assez empirique.
Un autre question intéressante est la robustesse du système quand l'utilisateur ferme les yeux, ouvre la bouche, ou fait quelque chose que le modèle synthétique ne peut actuellement reproduire : en général, le système n'est pas perturbé, parce qu'il suit assez de points d'intérêt dans le visage pour se permettre d'avoir des observations imprécises sur quelques uns d'entre eux. Pour rendre le système encore plus robuste dans de tels cas, il conviendrait de mettre à jour les motifs synthétisés non seulement du point de vue de la position et de l'orientation du sujet par rapport à la caméra, mais aussi du point de vue du contenu, lié aux expressions faciales du sujet. Nous disposons déjà de premiers résultats concernant la synthèse d'expressions faciales sur les clones, et nous travaillons actuellement sur leur intégration dans la boucle d'analyse/synthèse, à la fois pour mieux modéliser le locuteur durant la phase d'analyse du site émetteur, et retranscrire ses expressions faciales sur les clones des sites récepteurs.

3 Spatialisation vidéo

Le deuxième aspect du traitement d'images que nous étudions dans le but de créer un espace de conférence virtuel est la *spatialisation vidéo* pour le contrôle des images de fond de la scène, représentée uniquement par quelques vues 2D réelles mais non calibrées et sans modèle CAO 3D explicite. Un tel processus doit pouvoir nous offrir la possibilité de visualiser la salle de réunion en question depuis n'importe où et dans n'importe quelle direction, au lieu d'imposer un point de vue unique pour chaque site participant, comme le font les systèmes de téléconférence actuels. Cependant, il semble impossible, en termes d'acquisition d'une part et de liaisons bas-débit d'autre part, de créer dans un premier temps puis de transmettre toutes les vues nécessaires de la scène. Notre travail utilise donc la trilinéarité combinée au plaquage de texture pour compresser les données à transmettre et accroître l'information en créant des points de vues inédits.
Pour ce faire, nous nous appuyons sur une première méthode de base permettant la reconstruction d'une vue existante à partir de deux vues voisines (résumée en section 3.1). Cette méthode de reconstruction constitue aujourd'hui une étape de validation de l'utilisation de la trilinéarité, que nous avons ensuite étendue à la synthèse de vues inexistantes (section 3.3) afin de couvrir plus largement l'espace virtuel.

3.1 Régénération de vues réelles

Par extension des concepts de stéréovision à trois vues perspectives de la même scène (voir figure 5), nous définissons les *tenseurs trilinéaires*, qui peuvent être exprimés en termes algébriques par quatre systèmes trilinéaires modélisés pour la première fois par A. Shashua [17] et établissant quatre jeux de dix-huit paramètres qui déterminent parfaitement la configuration spatiale des trois caméras initiales. Une seule des quatre formes trilinéaires est proposée ici (équation 3), la définition des autres formes ainsi que l'expression analytique complète des quatre vecteurs associés ((α_i) pour la forme proposée) en fonction des paramètres intrinsèques et extrinsèques des caméras en place sont détaillées dans [18, 19].

$$\begin{cases} x'(\alpha_1 x'' + \alpha_2 y'' + \alpha_3) + x' x(\alpha_4 x'' + \alpha_5 y'' + \alpha_6) x(\alpha_7 x'' + \alpha_8 y'' + \alpha_9) + \alpha_{10} x'' + \alpha_{11} y'' + \alpha_{12} = 0 \\ y'(\alpha_1 x'' + \alpha_2 y'' + \alpha_3) + y' x(\alpha_4 x'' + \alpha_5 y'' + \alpha_6) + x(\alpha_{13} x'' + \alpha_{14} y'' + \alpha_{15}) + \alpha_{16} x'' + \alpha_{17} y'' + \alpha_{18} = 0 \end{cases} \quad (3)$$

En utilisant l'une des formes trilinéaires [20], nous pouvons reconstruire une vue existante à partir de deux vues voisines par l'algorithme suivant (figure 9(a)):

– une phase d'analyse, utilisant un jeu de points homologues dans trois vues originales non calibrées permet d'obtenir une estimation des dix-huit paramètres d'une forme trilinéaire (pour plus de détails concernant la définition des paramètres trilinéaires, se reporter à [21]);

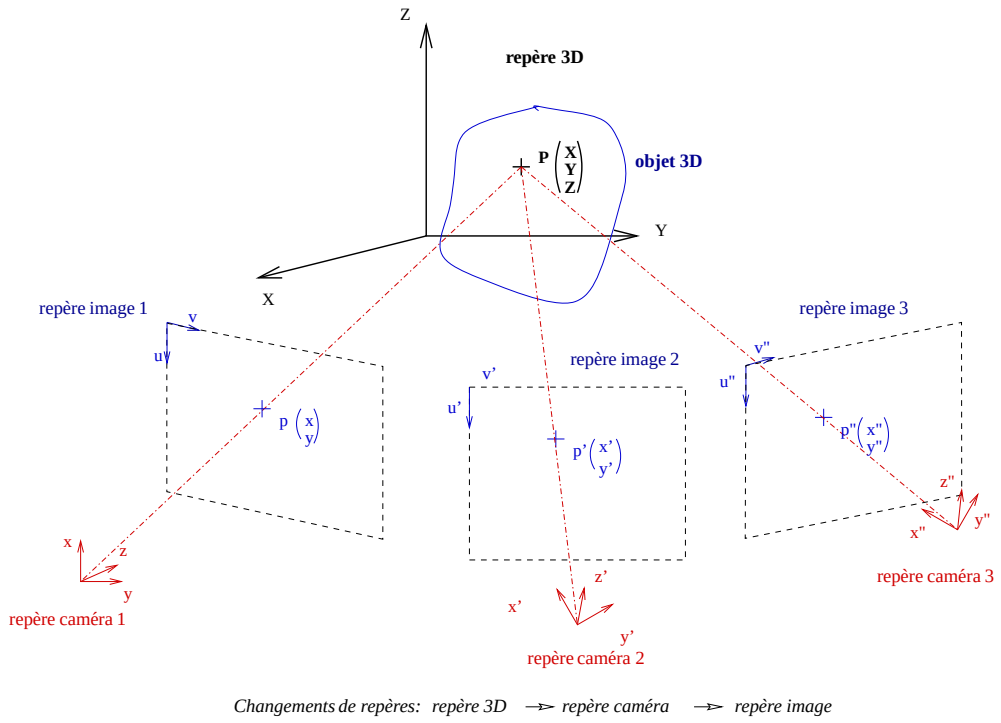


FIG. 5 – *Configuration initiale des caméras.*

– une phase de synthèse, utilisant les correspondants de deux des images initiales et les paramètres précédemment estimés, permet de reconstruire la troisième image.

Cette technique était initialement “orientée pixel”, mais il nous a paru nécessaire de la modifier pour en faire une méthode “orientée maillage” car telle quelle, elle présente plusieurs inconvénients concernant le temps de calcul et la qualité visuelle des résultats. En effet la méthode initiale nécessite des mises en correspondance denses tant au niveau de la phase d’analyse que de la phase de synthèse, ces processus sont trop longs pour envisager par la suite d’utiliser cet algorithme de resynthèse en temps-réel. De plus, les reconstructions “orientées pixels” obtenues sont incomplètes et nécessitent de nombreux post-traitements afin d’obtenir une reconstruction 2D visuellement satisfaisante. Dans le cas de la méthode “orientée maillage”, un maillage basé sur les points d’intérêt des images est associé à chaque image originale, à la suite de la phase d’analyse. La séquence de trois images initiales est alors remplacée par une texture et trois maillages plus ou moins denses suivant la complexité de la scène. La phase de synthèse n’utilisent plus alors que les noeuds des maillages de deux images originales et les dix-huit nombres flottants préalablement estimés pour reconstruire le maillage associé à la troisième image, avant plaquage de la texture de référence (figure 6).
L’information initiale concernant les deux vues originales à transmettre, pour reconstruire la troisième, est donc réduite à une texture complète accompagnée de deux maillages téléchargés à l’avance. En termes de données, un jeu de dix-huit nombres flottants remplace une vue complète (la vue à resynthétiser), mais il faut évidemment procéder à la reconstruction de cette vue au niveau des sites récepteurs.

3.2 Optimisation des maillages de base par mosaïcing

La méthode orientée maillage présentée ci-dessus nécessite de définir des maillages originaux associés à une texture de base. Ces maillages dépendent de la zone commune entre les trois vues de la séquence originale utilisée. Si les trois vues originales ont une zone commune très restreinte, le nombre de points d’intérêt caractéristiques apparaissant à la fois dans les trois vues

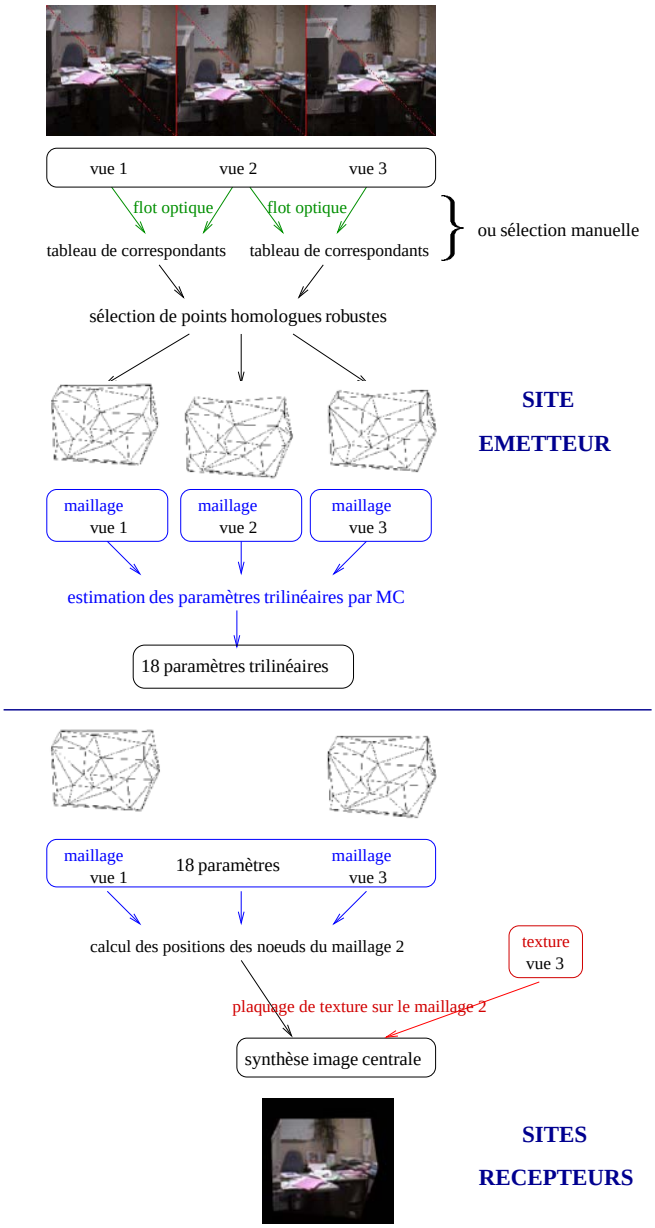


FIG. 6 – *Méthode "orientée maillage"*

est lui aussi très restreint et les trois maillages associés ne couvriront donc pas une zone large. Or, si les maillages utilisés sont peu étendus, la vue reconstruite le sera également puisqu'en réalité, elle est elle-même constituée d'un maillage recalculé sur lequel est plaquée une texture originale. Il semble donc intéressant d'étendre au maximun les maillages avant d'utiliser les algorithmes d'estimation de paramètres trilinéaires et de resynthèse de vues. En utilisant le principe des mosaïques d'images [22] nous étendons les maillages originaux à des zones non communes aux trois vues initiales.

En effet, à partir de trois vues nous définissons trois maillages basés sur les points caractéristiques de la zone commune aux trois vues. Ces trois maillages et l'une des vues utilisée comme texture, par exemple la troisième, sont les entrées nécessaires au prétraitement que constitue le mosaïcing. En déterminant les homographies de passage de la troisième vue à la seconde et à la première, puis en ajoutant au maillage correspondant à l'image 3, prise comme texture de référence, *n* points situés sur les bords de cette image 3 et enfin en appliquant à ce nouveau maillage les transformations homographiques précédemment déterminées nous obtenons

deux nouveaux maillages contenant eux aussi *n* points supplémentaires. Ainsi les trois maillages qui seront utilisés par la suite ne dépendent plus uniquement de la zone commune aux trois vues initiales, mais ont pu être étendues à la zone de couverture de la vue 3 (figure 7). Les maillages 1 et 2 pourront en particulier contenir des noeuds représentant des points caractéristiques initialement non présents respectivement dans les vues 1 et 2.

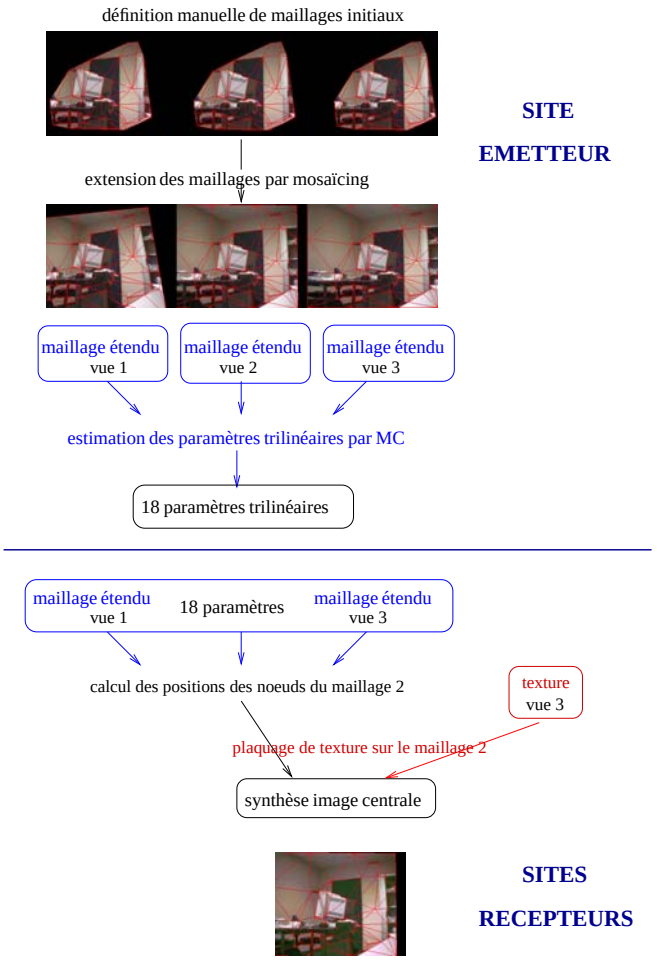


FIG. 7 – *Méthode "orientée maillage" sur des vues étendues par mosaïcing*

Etant en mesure de reconstruire une vue à partir de ses voisines en nous basant sur des maillages peu restrictifs, nous nous intéressons maintenant à la synthèse de vues a priori inexistantes.

3.3 Synthèse de vues virtuelles

Des extensions possibles de la méthode de reconstruction ont été étudiées afin de créer, à partir d'un jeu de vues initiales, des points de vue virtuels simulant un changement de distance focale ou une transformation géométrique 3D de la caméra relative au point de vue à reconstruire [23]. En appliquant directement sur les paramètres trilinéaires les modifications algébriques simulant des changements de paramètres intrinsèques (distance focale) ou extrinsèques (position et orientation 3D) de la caméra relative à la reconstruction, nous pouvons générer de nouvelles vues. Ces modifications algébriques sont résumées en figure 8 (pour plus de détails se reporter à [18, 19]). Après avoir modifié le vecteur de 18 paramètres, seule l'étape de synthèse de la méthode est nécessaire pour définir un point de vue à priori inconnu (figure 9). Quelques résultats visuels sont présentés figures 10 et 11; ils nous permettent d'envisager une couverture quasiment globale de l'espace de réunion à partir de quelques vues de référence correctement choisies.

	Paramètres trilinéaires	Foc.	Rot.
changement de focale	$\begin{cases} \alpha'_i = \alpha_i & i = 1..6 \\ \alpha'_i = c.\alpha_i & i = 7..18 \end{cases}$		
rotation de caméra	$\begin{cases} \alpha'_1 = c_\eta \alpha_1 - \frac{s_\eta \alpha_{16}}{k_v^2 f^2} \\ \alpha'_2 = c_\eta \alpha_2 - \frac{s_\eta \alpha_{17}}{k_v^2 f^2} \\ \alpha'_3 = c_\eta \alpha_3 - \frac{s_\eta \alpha_{18}}{k_v^2 f^2} \\ \alpha'_4 = c_\eta \alpha_4 - \frac{s_\eta \alpha_{13}}{k_v^2 f^2} \\ \alpha'_5 = c_\eta \alpha_5 - \frac{s_\eta \alpha_{14}}{k_v^2 f^2} \\ \alpha'_6 = c_\eta \alpha_6 - \frac{s_\eta \alpha_{15}}{k_v^2 f^2} \\ \alpha'_i = \alpha_{i, i=7..12} \\ \alpha'_{13} = c_\eta \alpha_{13} + k_v^2 f^2 s_\eta \alpha_4 \\ \alpha'_{14} = c_\eta \alpha_{14} + k_v^2 f^2 s_\eta \alpha_5 \\ \alpha'_{15} = c_\eta \alpha_{15} + k_v^2 f^2 s_\eta \alpha_6 \\ \alpha'_{16} = c_\eta \alpha_{16} + k_v^2 f^2 s_\eta \alpha_1 \\ \alpha'_{17} = c_\eta \alpha_{17} + k_v^2 f^2 s_\eta \alpha_2 \\ \alpha'_{18} = c_\eta \alpha_{18} + k_v^2 f^2 s_\eta \alpha_3 \end{cases}$ $c_\eta = cos(\eta) \quad s_\eta = sin(\eta)$ η = angle de rotation $k_v^i f^i$ paramètre interne relatif aux caméras	×	
translation de caméra	$\begin{cases} \alpha'_7 = \alpha_7 + k_u^2 f^2 r_{31}^1 c \\ \alpha'_8 = \alpha_8 + k_u^2 f^2 r_{32}^1 c \\ \alpha'_9 = \alpha_9 + k_u^2 f^2 r_{33}^1 c \\ \alpha'_{10} = \alpha_{10} + k_u^1 f^1 k_u^2 f^2 r_{11}^1 c \\ \alpha'_{11} = \alpha_{11} + k_u^1 f^1 k_u^2 f^2 r_{12}^1 c \\ \alpha'_{12} = \alpha_{12} + k_u^1 f^1 k_u^2 f^2 r_{13}^1 c \\ \alpha'_i = \alpha_{i, i=1...6, 13...18} \end{cases}$ r_{jk}^i paramètre relatif à la position des caméras	×	×

FIG. 8 – $(\alpha_i)_{i=1..18}$ *sont les paramètres trilinéaires initiaux et* $(\alpha'_i)_{i=1..18}$ *les paramètres modifiés. Les manipulations de paramètres (liés aux changements de point de vue souhaités) sont classées ici par ordre croissant de difficulté.*

Ceci est particulièrement intéressant pour notre application. Effectivement, après une phase de téléchargement de quelques vues réelles non calibrées de l'espace de réunion et une pré-estimation des vecteurs de paramètres trilinéaires correspondants, nous sommes capables par calculs algébriques de créer, pour chaque site indépendamment des autres, de nouveaux points de vue cohérents pour chaque participant, basés sur ses paramètres de mouvement (rotation et translation globales de sa tête et de ses yeux) et sur sa position virtuelle dans la salle de réunion. Parmi les perspectives du projet, il reste bien sûr à définir une stratégie de couverture complète de l'espace de réunion afin de connaître le nombre suffisant de maillages et de textures à télécharger sur chaque site récepteur pour obtenir une restitution acceptable et cohérente des images de fond pour chaque utilisateur.

4 Remarques concluantes

L'imagerie virtuelle offre de nouvelles perspectives en ce qui concerne les systèmes de téléconférence, qui utilisent des liaisons très bas-débit [24]. Après une phase préliminaire de téléchargement (i.e. transmission des modèles CYBERWARE des participants et de plusieurs

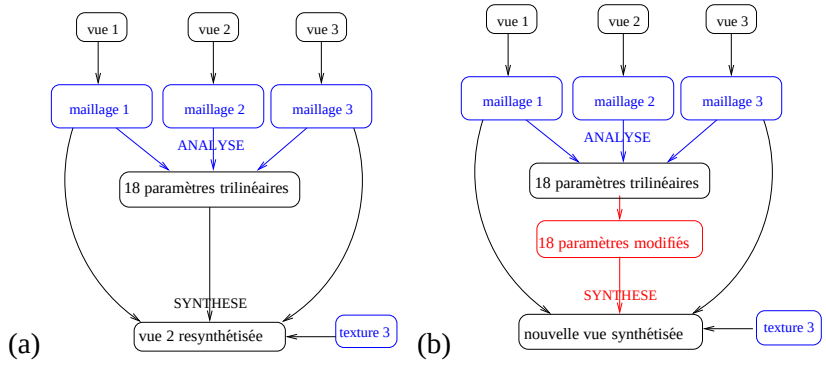


FIG. 9 – *Méthodes de reconstruction de vues réelles (a) et de synthèse de vues virtuelles (b).*

points de vue de la salle de réunion), de tels systèmes peuvent offrir plus de liberté d'interaction qu'une réunion classique; par exemple, les participants peuvent choisir leur point de vue par rapport à leur place virtuelle dans la salle de réunion ou par rapport à leurs centres d'intérêt.
Dans l'état actuel du projet, des prototypes d'algorithmes de clonage et de spatialisation ont été développés sur des stations de travail SGI utilisant la librairie graphique OpenGL. Les exemples de la figure 2 sont extraits d'une séquence vidéo d'une durée de 30 secondes acquise à la cadence de 10 images de résolution 320×242 par seconde. La bande passante nécessaire (hors téléchargement) pour transmettre les paramètres globaux au visualiseur de scènes virtuelles est 6 paramètres/image $\times$ 2 octets/paramètre $=$ 12 octets/image. Ces 6 paramètres/image sont également utilisés pour resynthétiser au niveau de chaque poste récepteur le point de vue de l'utilisateur sur l'espace de réunion commun. La gestion des images de fond en séance ne nécessite donc aucune nouvelle transmission d'information, mais uniquement des traitements basés sur les paramètres de position de la personne visualisant la scène à chaque site récepteur.
Nous pouvons dès maintenant imaginer le déroulement d'une visioconférence virtuelle réunissant quatre participants autour d'une table commune (figure 12), chacun d'entre-eux ayant son propre point de vue sur l'espace de réunion et sur les autres participants (figure 13). Ceci nous amène à la problématique plus générale d'intégration d'objets $3D$ (les clones dans le cas de notre application) dans des images $2D$ (les vues reconstruites de l'espace de réunion). Ce problème est très ouvert et constitue dans une large mesure les perspectives de nos futurs travaux, en plus de la restitution des expressions faciales des participants et de la définition d'un pavage complet de l'espace de réunion.
Une autre phase future du projet concerne l'intégration de tels outils dans un système réseau standard et dans un browser WEB sur Internet. Deux solutions sont envisagées : la première consiste à implanter un module de téléconférence virtuelle parmi les outils MBone multicast existants [2], la difficulté à prévoir venant ici de la disponibilité et des performances de la librairie OpenGL sur différentes plate-formes matérielles ; la seconde, actuellement retenue, serait l'utilisation des algorithmes de synthèse via le langage VRML [25], le challenge ici étant de contourner les problèmes posés par la communication multi-points temps-réel entre les différents participants via Internet.
Enfin, l'essor des réseaux multimédia mobiles pourrait offrir à l'avenir un nouvel espace d'utilisation de ce projet, en distinguant les deux modes suivants : le téléchargement via des réseaux fixes et le mode de fonctionnement en séance sur des terminaux multimédias via des réseaux mobiles.
Dans cet article, nous avons présenté plusieurs outils de traitements vidéo qui, dans le cadre d'un système complet de téléconférence virtuelle, devront être associés à des outils de traitement audio comme ceux développés par l'IRCAM en spatialisation du son [26]. En outre, nous restons attentifs aux évolutions du groupe MPEG-4 SNHC [27], dont le but est d'encoder efficacement

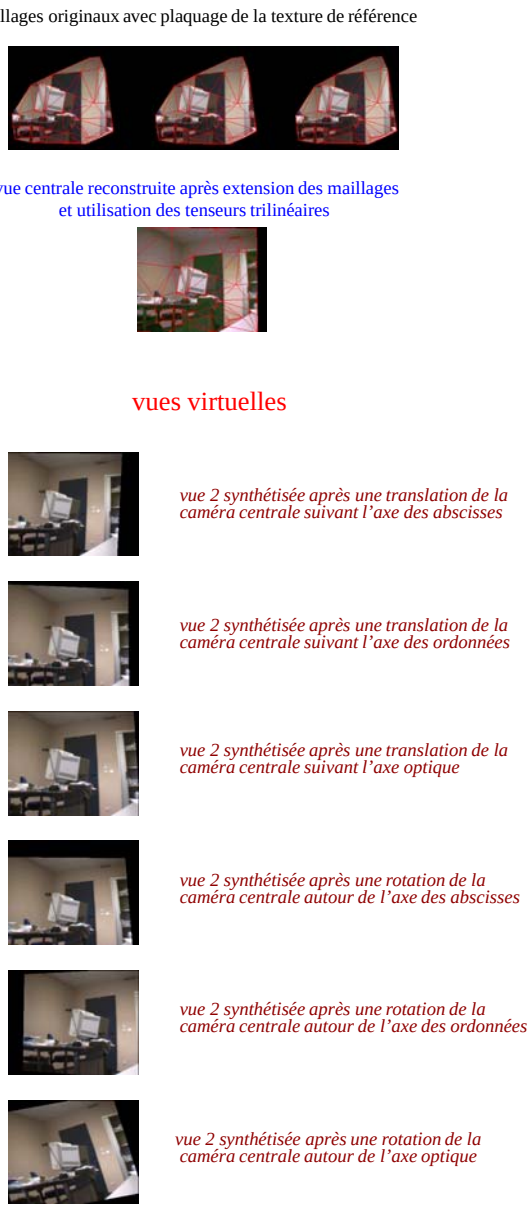


FIG. 10 – *Synthèse de vues inédites*

des environnements interactifs $2D$ et $3D$ mélangeant audio et vidéo temps réels avec des objets synthétiques. Nos approches concernant le clonage de visage et la restitution de l'espace virtuel s'inscrivent dans cette lignée.

Références

[1] IUT. Narrow-band visual telephone systems and terminal equipment, March 1996.

[2] Mbone (or ip multicast) Information Web. URL http://www.mbone.com.

[3] V. Jackobson and S. MacCanne. vat. Technical report, Lawrence Laboratory, University of California, Berkley, CA.

[4] IETF. Host extensions for IP multicasting, November 1988. rfc1112.

[5] H.-G. Musmann, M. Hötter, and Ostermann J. Object-oriented analysis-synthesis coding of moving images. *Signal Processing: Image Communication*, 1:117–138, 1989.

[6] A. Gagalowicz. Use of analysis/synthesis techniques for multimedia applications. Tutorials of the 1998 IEEE International Conference on Multimedia Computing and Systems, June 1998. Austin, Texas.

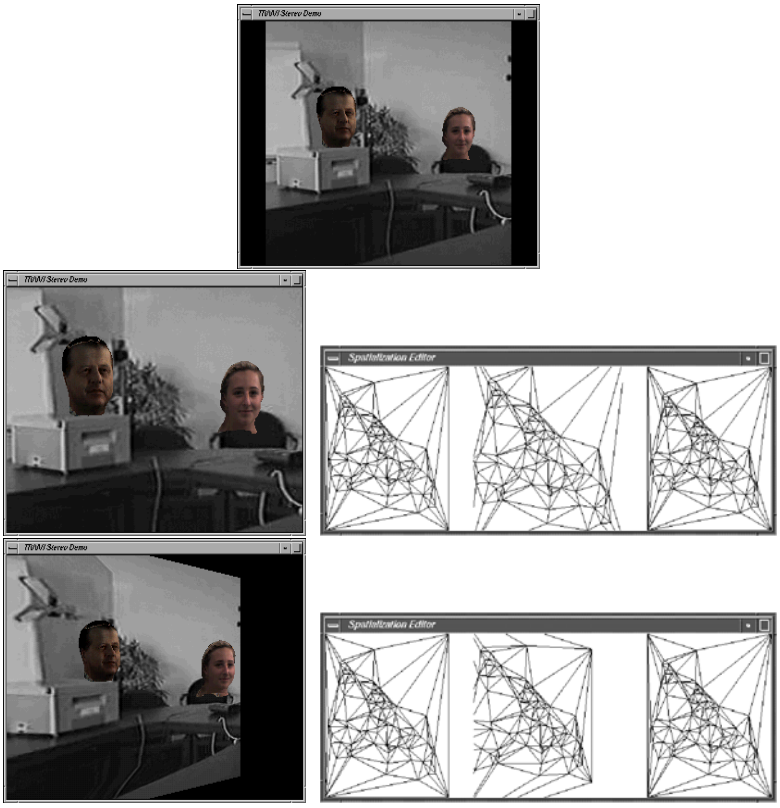


FIG. 11 – *Nouveaux points de vue synthétisés aprés un changement de distance focale ou une rotation de caméra. Les maillages des images de fond correspondants sont présentés à droite.*

[7] P.-E. Chaut, A. Sadeghin, A. Saulnier, and M.-L. Viaud. Création et animation de clones. In *Imagina — Méta–mondes/Metaverses*, pages 244–257, Monaco, Février 1997.

[8] D. Terzopoulos and K. Waters. Analysis and synthesis of facial image sequences using physical and anatomical models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6), June 1993.

[9] I. S. Pandzic, P. Kalra, and N. Magnenat Thalmann. Real time facial interaction. *Displays*, 15(3), 1995. *Butterworth — Heinemann*.

[10] I. A. Essa, S. Basu, T. Darrell, and A. Pentland. Modeling, tracking, and interactive animation of faces and heads using input from video. In *Computer Animation'96 Conference*, Geneva, Switzerland, June 1996.

[11] Cyberware home page. URL http://www.cyberware.com.

[12] S. Valente, J.-L. Dugelay, and H. Delingette. An analysis/synthesis cooperation for head tracking and video face cloning. In *Workshop on Perception of Human Action, ECCV Conference*, Freiburg, Germany, June 1998.

[13] S. Valente, J.-L. Dugelay, and H. Delingette. Geometric and photometric head modeling for facial analysis technologies. Technical report, Institut Eurécom, 1998.

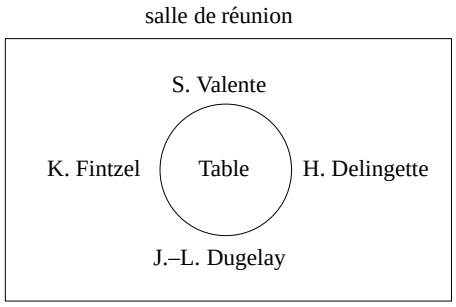


FIG. 12 – *Exemple de positions relatives des quatre participants à une visionconférence virtuelle.*

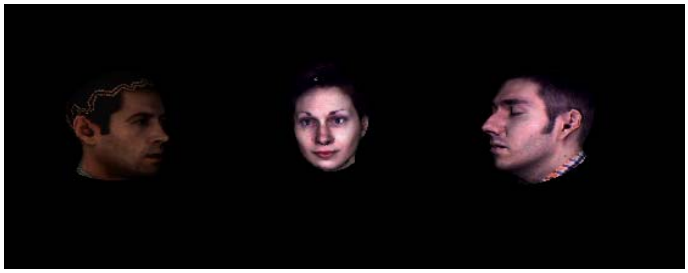


FIG. 13 – *Point de vue du participant H. Delingette sur les autres participants. Visualisation via l'interface CosmoWorld (VRML).*

[14] G. Hager and P. Belhumeur. Real–time tracking of image regions with changes in geometry and illumination. In *IEEE CVPR*, November 1996.

[15] Mpeg demo of the face tracking system. URL http://www.eurecom.fr/∼image/TRAIVI/valente-8points.mpg . (1782100 bytes).

[16] B. Bascle and A. Blake. Separability of pose and expression in facial tracking and animation. In *International Conference on Computer Vision*, Bombay, India, January 4-7 1998.

[17] A. Shashua. On geometric and algebraic aspect of 3D affine and projective structures from perspective 2D views. In J.-L. Mundy, A. Zisserman, and D. Forsyth, editors, *Applications of Invariance in Computer Vision*. Second European Workshop Invariants, Ponta Delagada, Azores, October 1993.

[18] K. Fintzel and J.-L. Dugelay. Défocalisation en spatialisation vidéo à partir de trois vues de référence (expressions analytiques). Technical report, EURECOM, Département Communications Multimédia, Sophia Antipolis, France, Février 1996.

[19] K. Fintzel and J.-L. Dugelay. Manipulations analytiques des paramètres trilinéaires pour la resynthèse d'images inédites. Technical report, EURECOM, Département Communications Multimédia, Sophia Antipolis, France, Novembre 1997.

[20] P. Bobet, J. Blanc, and R. Mohr. Aspect cachés de la trilinéarité. In *Proc. RFIA'96 Conf.*, pages 137–146, Rennes, France, Janvier 1996.

[21] S. Avidan and A. Shashua. Tensorial transfer: representation of N>3 views of 3D scenes. In *ARPA Image Understanding Workshop*, Palm Springs, CA USA, February 1996.

[22] O. Faugeras. De la géométrie au calcul variationnel: théorie et applications de la vision tridimensionnelle. In *RFIA'98*, pages 15–34, Clermont-Ferrand, France, Janvier 1998.

[23] K. Fintzel and J.-L. Dugelay. Spatialisation vidéo. In *Proc. CORESA'96 Conf.*, CNET Grenoble, France, Février 1996.

[24] J. Ohya, Y. Kitamura, F. Kishino, and N. Terashima. Virtual space teleconferencing: Real-time reproduction of tridimensional human images. *Journal of Visual Communication and Image Representation*, 6(1):1–25, March 1995.

[25] Vrml. URL http://vrml.sgi.com.

[26] J.-M. Jot. Synthesizing three–dimensional sound scenes in audio or multimedia production and interactive human–computer interfaces. In *L'Interface des Mondes Réels & Virtuels*, Montpellier, France, Mai 1996.

[27] MPEG–4 synthetic/natural hybrid coding. URL http://www.es.com/mpeg4-snhc/ .

Jean-Luc Dugelay né à Rouen en 1965, a rejoint l'institut Eurécom à Sophia Antipolis en 1992, où il s'occupe actuellement des activités d'enseignement et de recherche en image et vidéo au sein du département Communications Multimédia. Il effectua sa thèse de doctorat sur la Télévision stéréoscopique et l'estimation du mouvement $3D$ dans le département Traitement et Codage d'image avancés du CCETT (CNET/FRANCE TELECOM) à Rennes. Il reçu son diplôme de docteur de l'Université de Rennes en 1992. Ses activités se concentrent actuellement sur le traitement des signaux multimédia en particulier sur: les fractales et leurs applications au codage, tatouage et indexation d'images et les techniques d'imagerie $3D$ pour les systèmes de téléconférence virtuelle.

Katia Fintzel née en 1972 à Antibes, reçut en 1995 le diplôme d'ingénieur en informatique de l'Ecole Supérieure en Sciences Informatiques de l'Université de Nice Sophia-Antipolis. Ingénieur pour la société Espri Concept à Sophia Antipolis, elle poursuit actuellement sa thèse de Doctorat CIFRE au sein du laboratoire Communications Multimédia de l'institut Eurécom. Ces thèmes de recherche sont orientés vers les systèmes de communications vidéos, ils incluent la modélisation et le traitement d'image et plus particulièrement la spatialisation vidéo basée sur la théorie de la trilinéarité.

Stéphane Valente né en 1971, a commencé en 1996 une thèse de Doctorat au sein du Département Communications Multimedia de l'Institut Eurécom. Diplômé de l'Institut National des Télécommunications d'Evry en 1994, il a auparavant effectué son stage de fin d'études en reconnaissance de parole au Speech Technology Laboratory de Santa Barbara, Californie, et son service militaire en tant que Scientifique du Contingent en visualisation de données scientifiques au Commissariat à l'Energie Atomique à Bordeaux. Ses thèmes de recherche sont l'analyse et la synthèse d'images, le clonage de visages et la réalité virtuelle.

Hervé Delingette a reçu le diplôme d'ingénieur de l'Ecole Centrale de Paris en 1989 et le titre de docteur en sciences en 1994. Il est à présent chargé de recherche au sein du Projet Epidaure à l'INRIA Sophia-Antipolis. De 1990 à 1992 il a rejoint le laboratoire de vision de l'Université de Carnegie-Mellon à Pittsburgh (USA). En 1992, il a ensuite rejoint le laboratoire de recherche en Images de Synthèse de la société Nippon Telegraph and Telephone à Yokosuka (Japon). Ses travaux de recherche concernent l'imagerie médicale, la vision par ordinateur et la synthèse d'images.