

# SPATIALISATION VIDEO

katia FINTZEL  
Jean-Luc DUGELAY

Institut EURECOM  
Département Communications Multimédia  
route des Crêtes, BP 193, F-06904 Sophia Antipolis Cedex  
e-mail {fintzel,dugelay}@eurecom.fr  
url. <http://www.cica.fr/~image>

## 1 MOTIVATIONS

Cette étude s'inscrit dans le cadre d'un projet EURECOM de télé-virtualité: **TRAI**VI (TRAitement des Images Virtuelles), dont le but est à la fois de créer et de gérer des espaces virtuels de réunion. Nous voulons augmenter le degré de réalisme visuel de la télé-réunion, en permettant essentiellement la navigation dans la salle de réunion virtuelle, via la resynthèse de vues existantes ou éventuellement inexistantes de cette scène, dont on ne possède pas de modèle 3D proprement dit.

Nous pouvons associer un aspect de compression "cachée" à la solution que nous apportons à un tel problème, une solution qui peut également contribuer à d'autres applications, telles que "Virtual Walk", dont l'objectif est de donner l'illusion de se promener dans un espace, alors que l'on ne dispose que de quelques vues réelles.

## 2 PRINCIPE ET METHODE

### 2.1 LA TRILINEARITE

#### 2.1.1 Justification de l'utilisation de la trilinearité

L'objectif avancé est la resynthèse d'une vue existante d'une scène 3D à partir d'autres points de vue de cette même scène. Nous savons, par la géométrie épipolaire définie en vision binoculaire, que si on possède deux vues de référence d'une même scène 3D, tout point  $m_1$  projeté d'un point  $M$  de la scène dans la première vue admet un correspondant  $m_2$  (où se projetterait  $M$  dans la seconde vue) situé sur la droite épipolaire  $ep_2(m_1)$  de  $m_1$  dans la vue 2 [OF88]. En utilisant deux fois cette propriété sur un triplet d'images prises deux à deux, il est clair que l'on définit précisément la place du correspondant dans la troisième vue d'un couple de points homologues des deux premières vues: c'est la **contrainte trinoculaire**. (figure 1)

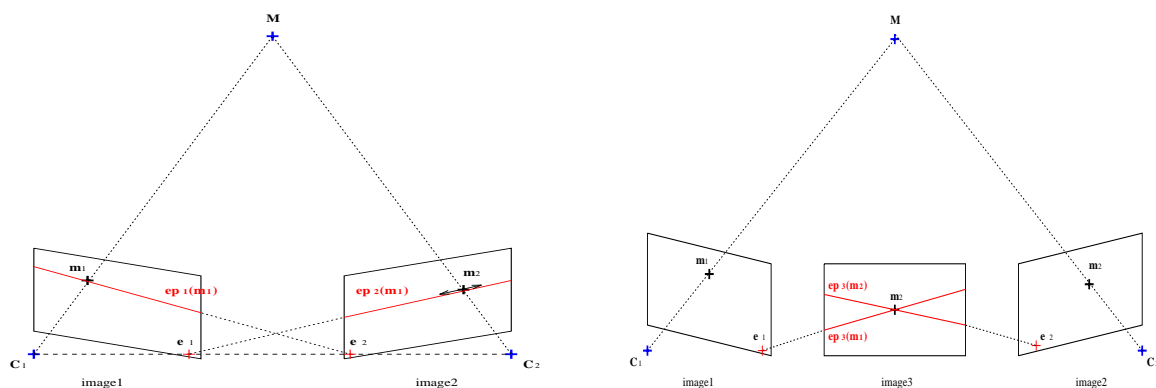


figure 1

## 2.1.2 Système trinéaire de A. SHASHUA

De la contrainte précédente découle l'idée qu'il existe une expression complètement définie de l'ensemble des points d'une vue de scène 3D, obtenue à partir de deux autres vues de cette même scène. Cette expression avancée par A. SHASHUA est donnée sous forme d'un **système de deux équations trinéaires**: [AS94]

$$\begin{cases} x'(\alpha_1 x'' + \alpha_2 y'' + \alpha_3) + x'x(\alpha_4 x'' + \alpha_5 y'' + \alpha_6) + x(\alpha_7 x'' + \alpha_8 y'' + \alpha_9) + \alpha_{10} x'' + \alpha_{11} y'' + \alpha_{12} = 0 \\ y'(\beta_1 x'' + \beta_2 y'' + \beta_3) + y'x(\beta_4 x'' + \beta_5 y'' + \beta_6) + x(\beta_7 x'' + \beta_8 y'' + \beta_9) + \beta_{10} x'' + \beta_{11} y'' + \beta_{12} = 0 \end{cases}$$

avec  $\alpha_i = \beta_i$  pour  $i = 1 \dots 6$

que l'on réorganise, après avoir fixé  $\alpha_{12}$  à 1, en:

$$\begin{cases} x' = -\frac{x(\alpha_7 x'' + \alpha_8 y'' + \alpha_9) + \alpha_{10} x'' + \alpha_{11} y'' + 1}{(\alpha_1 x'' + \alpha_2 y'' + \alpha_3) + x(\alpha_4 x'' + \alpha_5 y'' + \alpha_6)} \\ y' = -\frac{x(\alpha_{13} x'' + \alpha_{14} y'' + \alpha_{15}) + \alpha_{16} x'' + \alpha_{17} y'' + \alpha_{18}}{(\alpha_1 x'' + \alpha_2 y'' + \alpha_3) + x(\alpha_4 x'' + \alpha_5 y'' + \alpha_6)} \end{cases}$$

## 2.2 ALGORITHME DE RESYNTHESE

Ces principes théoriques de base ayant été établis, notre méthode de resynthèse se constitue de deux étapes:

- La première phase consiste, à partir de trois vues d'une même scène 3D, à estimer les paramètres précédents par la méthode des Moindres Carrés, nécessitant deux mises en correspondance denses obtenues par calcul de flots optiques entre les images de référence [BC&JLD&DP95]. (*figure 2*)

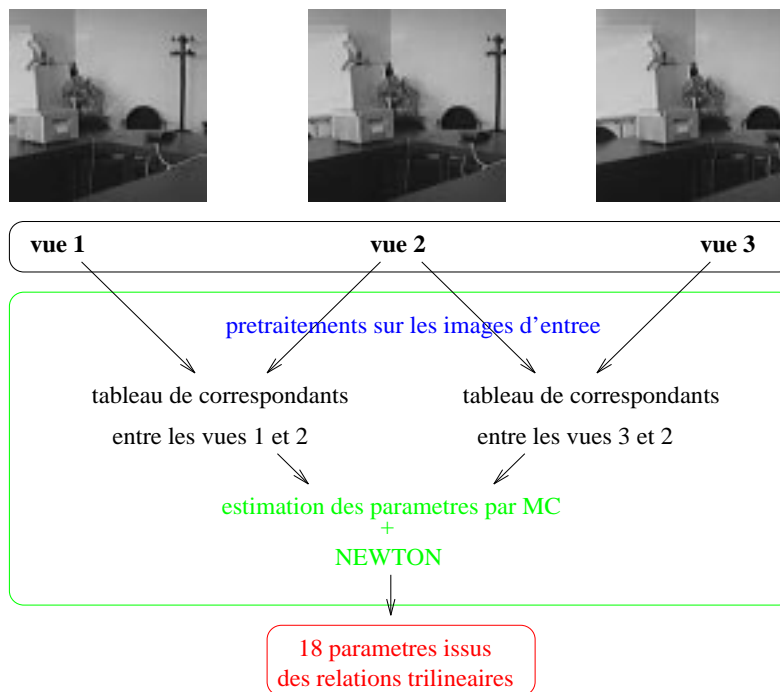


figure 2

- La seconde étape constitue véritablement la phase de reconstruction de l'une des trois vues à partir des deux autres, elles-même mises en correspondance, et des paramètres précédents. (*figure 3*)

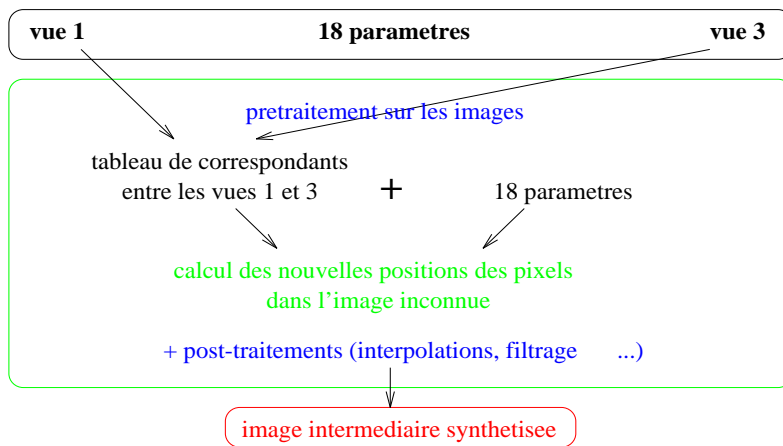


figure 3

### 2.3 PREMIERS RESULTATS

Un autre exemple d'image resynthétisée par la méthode présentée ci-dessus est exposé ici (figure 4). Notons que l'on peut considérer cette technique en termes de compression, puisque dix-huit paramètres associés à deux images nous permettent clairement de restituer trois images, ce qui représente un potentiel non négligeable et croissant avec la taille des images de référence.

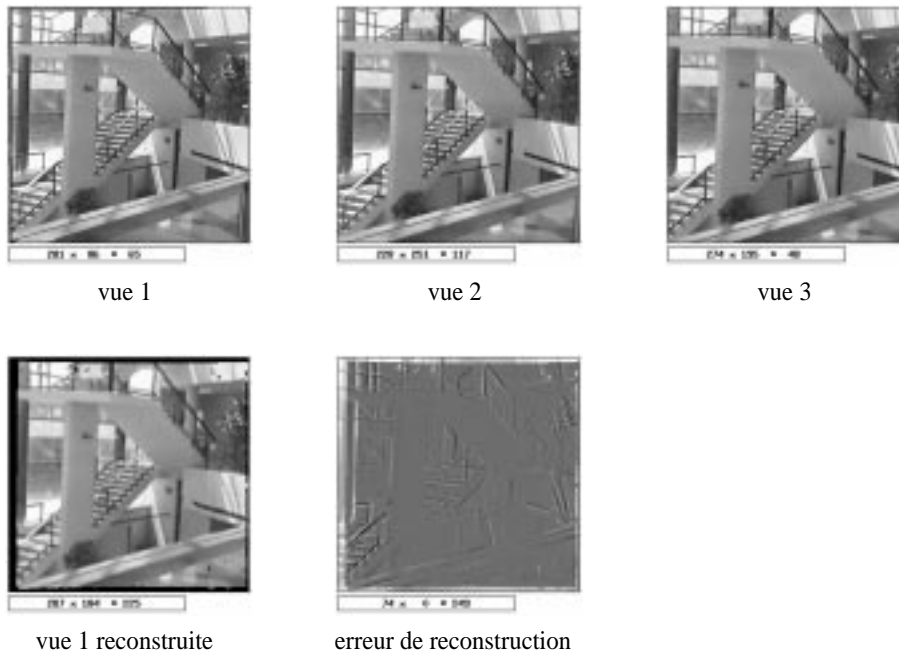


figure 4

Nous avons implanté la méthode présentée suivant nos propres choix techniques, en particulier pour ce qui concerne le matching des vues de référence et l'estimation des paramètres. Cependant la théorie relative aux équations trilinéaires précitées avait déjà fait l'objet de mises en oeuvre [JB&RM95] utilisant d'autres moyens et dans le cadre d'applications différentes. Aussi la méthode décrite ne constitue dans un premier temps que la validation de la théorie utilisée, une validation certes nécessaire, très importante et à étendre dans le cadre de nos applications (pour reconstruire des vues inexistantes).

### 3 EXTENSIONS DE LA METHODE

#### 3.1 A PARTIR DE TROIS VUES

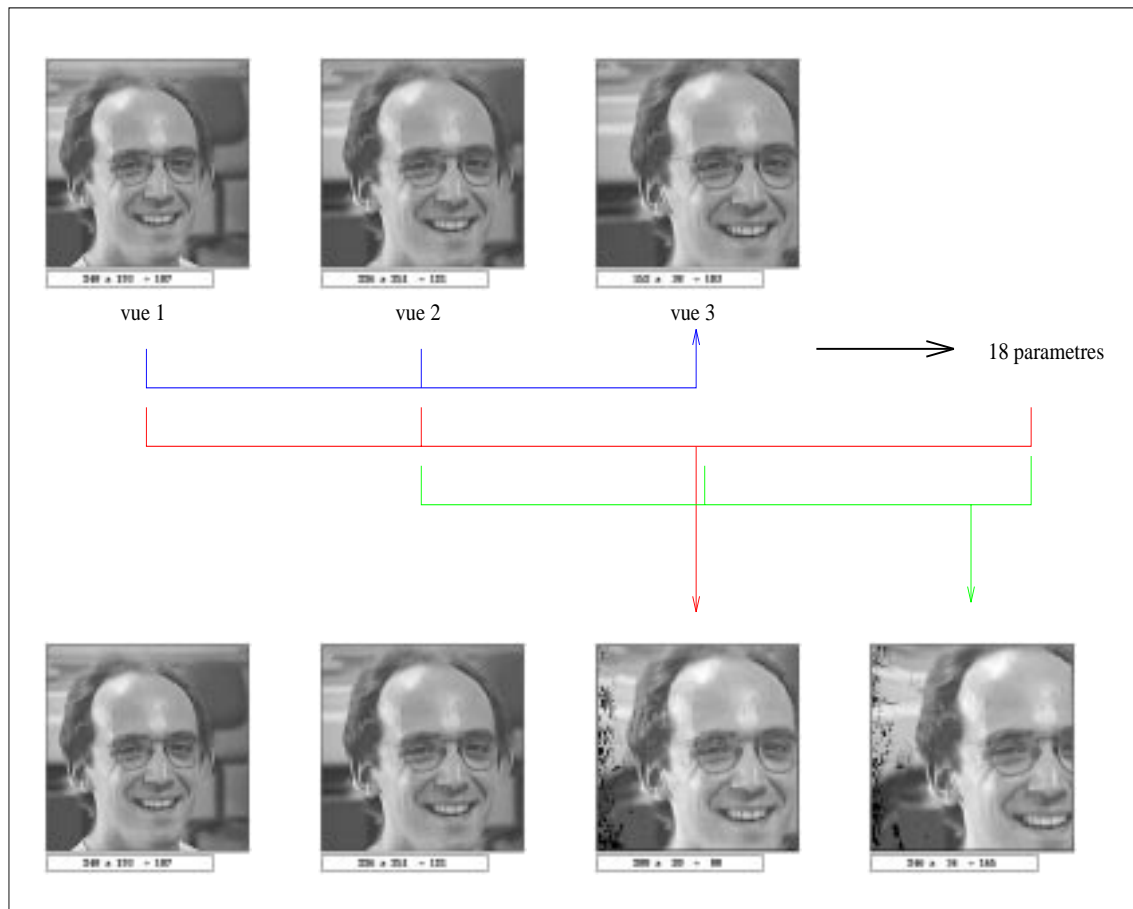
Nous nous plaçons ici dans le cas où nous utilisons toujours trois vues de la scène 3D. Il existe intuitivement deux grands types d'extensions de notre méthode, pour lesquelles nous agissons toujours lors de la phase de reconstruction de vues et jamais durant la première étape d'estimation des paramètres. En effet, nous pouvons soit modifier l'ordre des vues de référence lors de la phase de reconstruction, soit faire varier les paramètres estimés juste avant de les réutiliser pour une reconstruction.

##### 3.1.1 Redistribution des vues de référence lors de la reconstruction

L'idée avancée ici est de calculer les paramètres nécessaires pour la reconstruction de l'une des trois vues de référence en fonction des deux autres et de ne plus utiliser ces paramètres sur les vues appropriées, mais d'en changer l'ordre afin de reconstruire non pas l'image attendue mais une image inconnue, qui ne faisait pas partie des vues de référence.

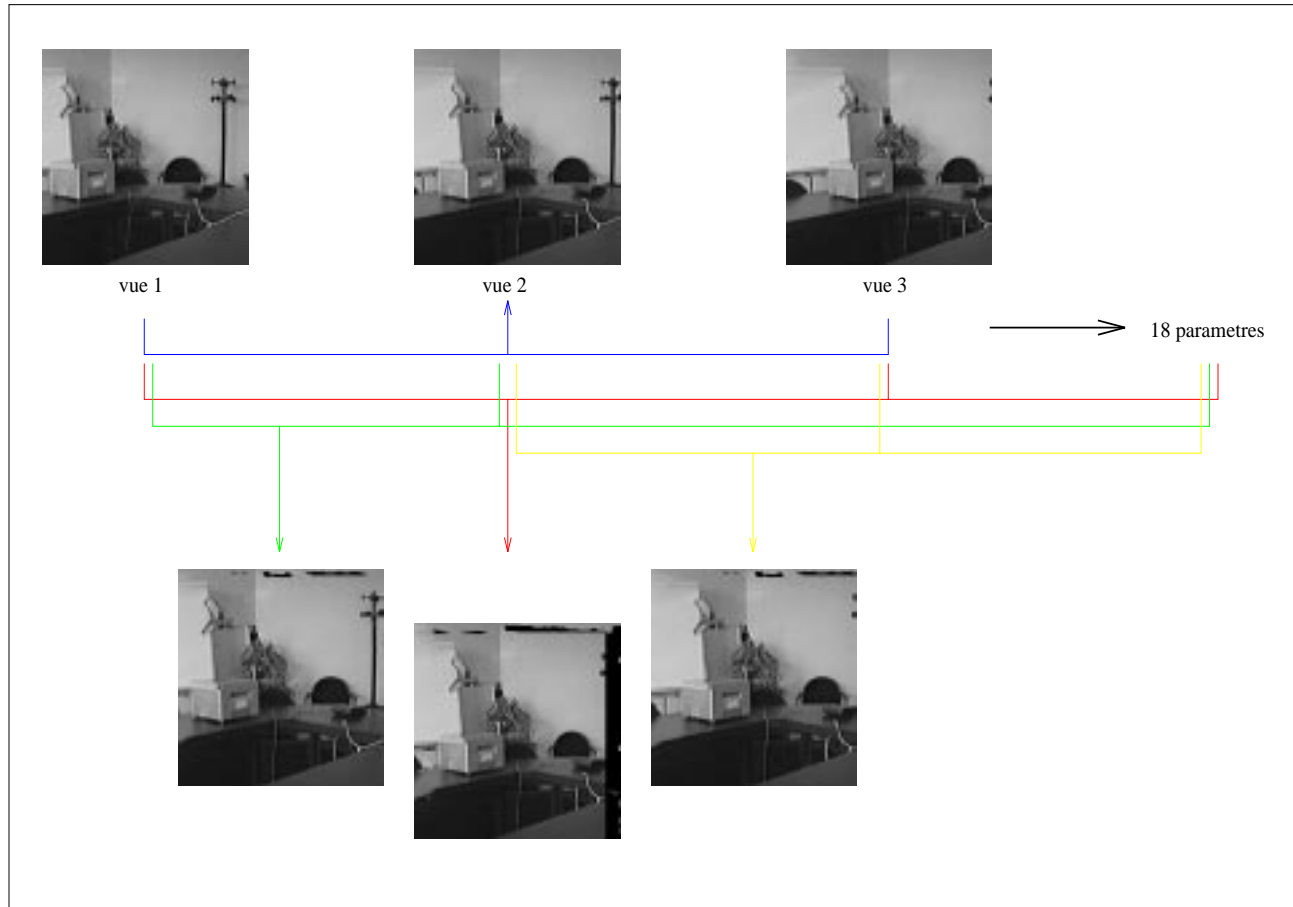
Typiquement nous pouvons donner ici deux exemples de nouvelles images synthétisées:

- Supposons que l'on ait trois vues de référence consécutives d'une scène 3D, dans un premier temps on cherche à estimer au mieux les paramètres qui nous permettent de reconstruire la troisième image à partir des images 1 et 2. Ensuite nous utilisons ces paramètres non pas associés aux images 1 et 2 mais aux images 2 et 3 pour reconstruire une nouvelle vue dans la continuité du mouvement relatif à la configuration des caméras. (*figure 5*)



*figure 5*

- On peut également essayer de reconstruire une image inconnue intermédiaire à deux vues de référence consécutives de la scène 3D. Pour cela on estime les paramètres nécessaires pour la resynthèse de l'image intermédiaire en fonction des images extérieures d'un triplet de vues, puis on associe par exemple ces dix-huit nombres aux images 1 et 2 au lieu des images 1 et 3. (*figure 6*)



*figure 6*

### 3.1.2 Variations de paramètres pour la phase de resynthèse

En faisant varier empiriquement les paramètres une fois précalculés, on peut également agir sur l'image reconstruite. On pourra par exemple observer des effets visuels de translation, de zoom ou de rotation sur les points de vue reconstruits, que l'on pourrait assimiler très grossièrement à des déplacements des caméras de référence. Dans le cadre de nos applications, ce type d'approche nous permettra de reconstruire des images visuellement cohérentes et qualitativement intéressantes (*figure 7*). Mais l'interprétation des perturbations dues à ces variations de paramètres sur les images reconstruites pose encore de nombreux problèmes. Un effet de translation en abscisses pourra par exemple être rendu par les variations suivantes:

$$\begin{cases} \alpha'_i = \alpha_i & \forall i \neq 12 \\ \alpha'_{12} = \lambda \alpha_{12} \end{cases}$$

où  $\lambda$  et  $\alpha_{12}$  définissent le sens de translation.

Pour une translation en y on aura:

$$\begin{cases} \alpha'_i = \alpha_i & \forall i \neq 18 \\ \alpha'_{18} = \lambda \alpha_{18} \end{cases}$$

Pour un effet de zoom:

$$\begin{cases} \alpha'_i = \alpha_i & \forall i = 1 \dots 6 \\ \alpha'_i = \lambda \alpha_i & \forall i = 7 \dots 18 \end{cases}$$

Enfin les variations suivantes permettront de simuler des rotations d'angle  $\theta$  suivant les abscisses, les ordonnées ou encore suivant l'axe de visée:

$$\left\{ \begin{array}{lll} \alpha'_i = \alpha_i \quad \forall i = 1 \dots 12 & \alpha'_i = \alpha_i \quad \forall i = 1 \dots 6 \text{ \& } 13 \dots 18 & \alpha'_i = \alpha_i \quad \forall i = 1 \dots 6 \\ \alpha'_{13} = \alpha_{13} \cos(\theta) - \alpha_4 \sin(\theta) & \alpha'_7 = \alpha_7 \cos(\theta) - \alpha_4 \sin(\theta) & \alpha'_7 = \alpha_7 \cos(\theta) - \alpha_{13} \sin(\theta) \\ \alpha'_{14} = \alpha_{14} \cos(\theta) - \alpha_5 \sin(\theta) & \alpha'_8 = \alpha_8 \cos(\theta) - \alpha_5 \sin(\theta) & \alpha'_8 = \alpha_8 \cos(\theta) - \alpha_{14} \sin(\theta) \\ \alpha'_{15} = \alpha_{15} \cos(\theta) - \alpha_6 \sin(\theta) & \alpha'_9 = \alpha_9 \cos(\theta) - \alpha_6 \sin(\theta) & \alpha'_9 = \alpha_9 \cos(\theta) - \alpha_{15} \sin(\theta) \\ \alpha'_{16} = \alpha_{16} \cos(\theta) - \alpha_7 \sin(\theta) & \alpha'_{10} = \alpha_{10} \cos(\theta) - \alpha_7 \sin(\theta) & \alpha'_{10} = \alpha_{10} \cos(\theta) - \alpha_{16} \sin(\theta) \\ \alpha'_{17} = \alpha_{17} \cos(\theta) - \alpha_8 \sin(\theta) & \alpha'_{11} = \alpha_{11} \cos(\theta) - \alpha_8 \sin(\theta) & \alpha'_{11} = \alpha_{11} \cos(\theta) - \alpha_{17} \sin(\theta) \\ \alpha'_{18} = \alpha_{18} \cos(\theta) - \alpha_9 \sin(\theta) & \alpha'_{12} = \alpha_{12} \cos(\theta) - \alpha_9 \sin(\theta) & \alpha'_{12} = \alpha_{12} \cos(\theta) - \alpha_{18} \sin(\theta) \\ & & \alpha'_{13} = \alpha_7 \sin(\theta) + \alpha_{13} \cos(\theta) \\ & & \alpha'_{14} = \alpha_8 \sin(\theta) + \alpha_{14} \cos(\theta) \\ & & \alpha'_{15} = \alpha_9 \sin(\theta) + \alpha_{15} \cos(\theta) \\ & & \alpha'_{16} = \alpha_{10} \sin(\theta) + \alpha_{16} \cos(\theta) \\ & & \alpha'_{17} = \alpha_{11} \sin(\theta) + \alpha_{17} \cos(\theta) \\ & & \alpha'_{18} = \alpha_{12} \sin(\theta) + \alpha_{18} \cos(\theta) \end{array} \right.$$

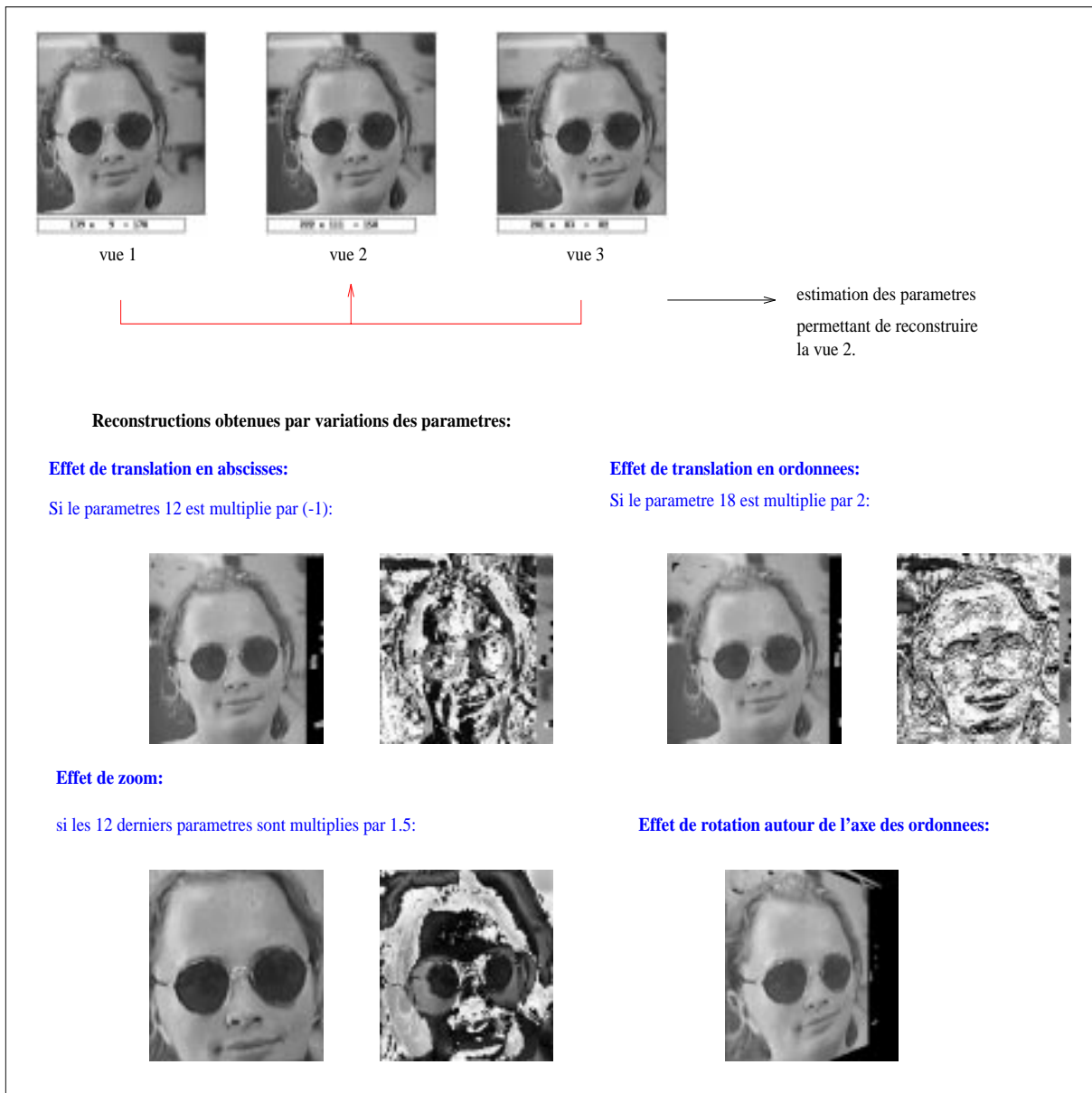


figure 7

### 3.2 SUR PLUS DE TROIS VUES

Afin de resynthétiser des points de vue inexistant, nous pouvons étendre la méthode de base en intégrant une quatrième vue parmi le jeu d'images de référence. Sur cette séquence de quatre vues de la même scène 3D, nous nous proposons de calculer deux jeux de paramètres permettant de reconstruire deux images différentes successives de la séquence toujours à partir des deux mêmes images de référence. Puis on reconstruit une série d'images inconnues, en utilisant toujours les deux images communes et un jeu de paramètres obtenu en faisant varier linéairement chaque paramètre entre les deux bornes définies dans les deux jeux préalablement calculés (figure 8).

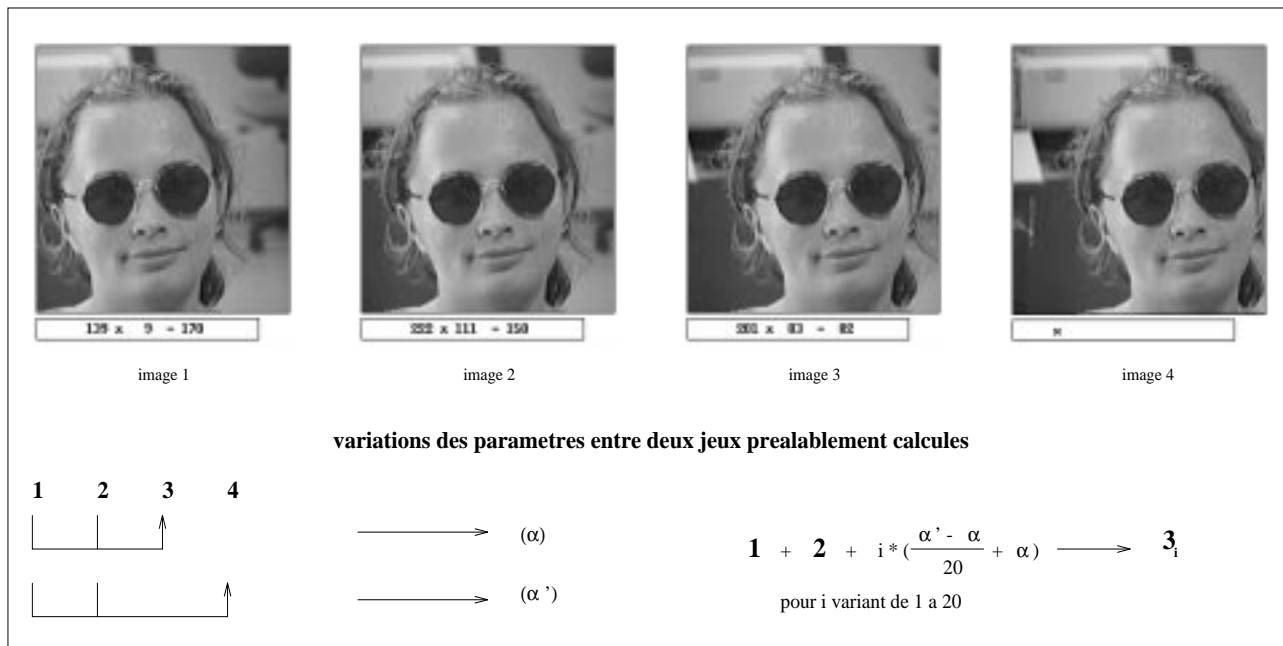


figure 8

Notons d'une part que la quatrième vue utilisée ici pourrait elle-même être une vue inconnue resynthétisée à partir des trois autres (cf §3.1.1) et d'autre part que nous pourrions encore intégrer de nouvelles vues de référence afin de prolonger le processus exposé ci-dessus, en prenant soin d'assurer des jonctions visuellement correctes.

### 3.3 VERIFICATIONS ANNEXES

#### 3.3.1 Robustesse des reconstructions

En marge du travail réalisé sur les valeurs des paramètres, issus des relations trilineaires, nous avons également essayé de comparer les reconstructions d'une même image de référence obtenues à partir de deux couples d'images de référence différents et des paramètres adaptés à chacun de ces couples. Les deux reconstructions s'avèrent cohérentes (figure 9), d'où l'idée d'utiliser plusieurs reconstructions pour rendre le point de vue resynthétisé le plus robuste possible.

#### 3.3.2 Pré-calcul d'un jeu de paramètres

Les dix-huit paramètres issus du système de relations trilineaires permettent d'exprimer sous une forme particulière les paramètres intrinsèques et extrinsèques des caméras qui ont servi à l'acquisition des trois vues différentes. L'idée de réutiliser ces paramètres pré-calculés sur un autre triplet d'images correctement choisies d'une autre scène 3D paraît intéressante et légitime en ce sens que les paramètres sont liés aux positions des caméras et non à la scène elle-même. (figure 9)

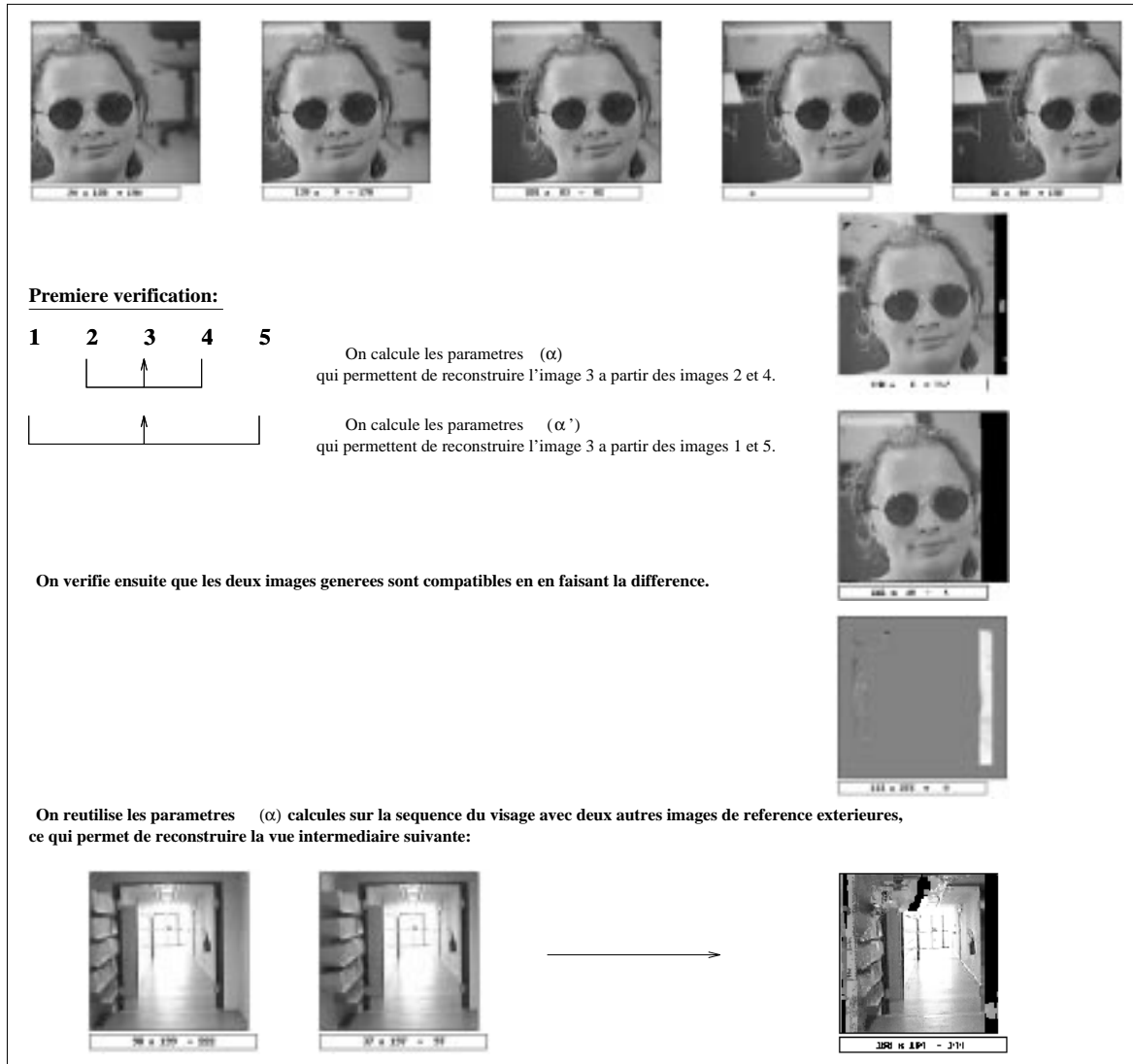


figure 9

## 4 CONCLUSION

Nous avons donc ébauché une solution qui permettrait, dans le cadre de nos applications, de s'affranchir de l'existence d'une vue pour envisager sa reconstruction à partir de plusieurs vues de référence. Pour systématiser la méthode, il sera nécessaire de maîtriser complètement les paramètres exprimés par le système trinéaire et de définir un "pavage" de l'espace qui nous intéresse. Rappelons enfin que suivant l'application visée, l'apport de cette étude en termes de compression pourra se révéler non négligeable (peu d'informations à priori permettant d'en reconstruire une quantité importante).

## Références

- [OF88] Olivier D. FAUGERAS, *Quelques pas vers la vision artificielle en trois dimensions*, article tiré de *Technique et Science Informatique*, INRIA Sophia-Antipolis septembre 1988.
- [RIH&RG&TC92] Richard I. HARTLEY, Rajiv GUPTA & Tom CHANG, *Stereo from Uncalibrated Cameras*, GE Corporate Research and Development IEEE 1992.
- [AS94] Amnon SHASHUA, *Projective structure from uncalibrated images: structure from motion and recognition*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1994.
- [BC&JLD&DP95] Bruno CHOQUET, Jean-Luc DUGELAY & Danielle PELE, *A Coding Scheme for Stereoscopic Television Sequences based on Motion Estimation-Compensation using a 3D Approach*, CCETT, EURECOM conférence IEE, 'IPA' UK 1995.
- [JB&RM95] Jérôme BLANC & Roger MOHR, *Calcul de vues de scènes 3D. Application à la compression vidéo*, LIFIA INRIA Grenoble, *journées sur les "Nouvelles Techniques pour la Compression et la Représentation des signaux audiovisuels"* Janvier 1995.