# Naviguer dans un monde virtuel

J. Blanc R. Mohr
Projet Movi, laboratoire Gravir [1]
Inria Rhône-Alpes
655, av. de l'Europe
38330 Montbonnot St Martin
E-mail : Jerome.Blanc@imag.fr

## 1 Introduction

**Participer à un film...** Imaginons que lors de la retransmission d'un film ou d'un événement sportif, le télespectateur puisse choisir librement son point de vue. Il pourrait se placer à côté d'un acteur, décider lui-même de ce qu'il veut voir de la scène, se situer au cœur de l'action, et participer ainsi à la réalisation. Le réalisateur choisit toujours les séquences, mais c'est bien le spectateur qui décide du découpage des plans à l'intérieur de chaque séquence.

Cette façon innovante de visualiser un film est une application de la *synthèse de nouveaux points de vues*, que nous allons présenter dans la suite de cet article.

Les applications envisagées sont en fait toutes celles des réalités virtuelles : exploration d'un terrain/cartographie, simulation, divertissements.

**Synthèse de points de vue** Nous regroupons sous cette appellation la "synthèse de nouvelles images d'une scène 3D sous un point de vue inédit". L'application que nous présentons est en effet la suivante : connaissant plusieurs vues d'une même scène tri-dimensionnelle, dites *vues de référence*, nous calculons d'autres vues de cette même scène. Pour ce faire il nous faudra bien sûr capter des informations de relief, aussi les vues de référence doivent être au minimum au nombre de deux. De plus, ces vues devront représenter la même scène, qui doit donc être statique. Pour les scènes animées, on peut effectuer le traitement sur chaque image (ou couple d'images) de la séquence.

**Les avantages de notre méthode**

- ▷ L'avantage par rapport aux méthodes classiques de synthèse d'images est que nous n'avons pas à créer un modèle numérique de la scène, opération longue et fastidieuse. Notre "modèle" est en fait compris dans la seule donnée des vues de référence.
- ▷ Un second avantage, qui en découle, est le plus grand réalisme des scènes synthétisées : elles sont composées à partir de vues réelles, et non de modèles mathématiques d'objets naturels.
- ▷ Enfin, nous nous plaçons toujours dans un cadre non-étalonné, ce qui autorise l'utilisation de caméras difficilement étalonnables, par exemple embarquées, autofocus, ou tenues à la main sans précaution particulière.

## 2 Méthode

Notre méthode est simple : après une phase d'appariement dense entre nos images de référence, nous procédons à une reconstruction de chaque point de la scène, que nous reprojetons ensuite sur le plan de la caméra virtuelle. Nous ne travaillons pour l'instant qu'à partir de deux images de référence.

1. Le projet Movi-Inria fait partie du laboratoire Gravir, Imag-Ujf-Cnrs.

**Un étalonnage est-il nécessaire ?** Il est inutile pour cette méthode d'utiliser des caméras étalonnées. En effet, nous pouvons à partir de deux images effectuer une reconstruction projective des points [Fau92]. Une projection perspective de ces points constituera l'image synthétisée.

Cependant, si nous voulons imprimer à la caméra virtuelle des mouvements euclidiens, plus familiers (se déplacer d'un certain nombre de mètres, pivoter d'un certain angle), nous devrons disposer d'un étalonnage au moins approximatif de la caméra. Nous montrerons que même une connaissance imprécise des paramètres constructeur de la caméra suffit à une bonne perception euclidienne de la scène virtuelle.

### 2.1 Appariement

La phase d'appariement est un préalable indispensable à la perception du relief. Nous avons choisi de n'apparier que des points, ce qui écarte d'emblée les problèmes rencontrés lors d'éventuelles phases de segmentation en contours ou en régions.

Pour apparier de nombreux points entre les deux images de référence, nous utilisons une technique de programmation dynamique [OK85, GLY92] hiérarchique, qui peut être couplée à une mesure corrélation robuste [LMR95]. Les cartes de disparité obtenues peuvent ensuite être régularisées avec d'autres méthodes, bien que dans la majorité des cas cela ne semble pas nécessaire pour notre application.

Comme elle constitue une grosse partie de nos résultats, nous détaillons notre méthode d'appariement à part, en section 3.

### 2.2 Reconstruction

De nos deux images de référence, nous pouvons extraire la matrice fondamentale $F_{12}$ [Fau92, BM95]. Nous fixons alors les paramètres intrinsèques $K$ de la caméra à des valeurs données, par exemple les paramètres constructeur, et nous obtenons la matrice essentielle $E_{12} = K^T.F_{12}.K$.

Nous calculons ensuite deux matrices de projection perspective dans les deux images $M_1$ et $M_2$, cohérentes avec $E_{12}$ [Luo92]. Après avoir aligné précisément les couples de points appariés sur des couples d'épipolaires conjuguées [HS94] afin de minimiser les erreurs, nous procédons simplement à une reconstruction aux moindres carrés. Nous obtenons point par point un modèle presque euclidien de la scène observée.

### 2.3 Projection

Le modèle obtenu est simplement projeté point par point sur le plan de la caméra virtuelle pour fabriquer la nouvelle image, *via* une matrice de projection perspective. On déplace la caméra en effectuant des transformations euclidiennes sur la matrice de projection.

## 3 Appariement

Après avoir développé un premier algorithme de mise en correspondance uniquement basé sur des mesures de corrélation, et que nous utilisions jusqu'à maintenant, nous avons voulu reposer le problème en termes de probabilités d'appariements.

Nous présentons ici cette nouvelle méthode, que nous commençons seulement à développer. Nous présenterons aussi quelques résultats préliminaires.

### 3.1 Notre but

**Données** Nous nous donnons deux images d'une même scène statique. Nous voulons obtenir des correspondances de points entre ces deux images. De ces images, nous pouvons calculer la géométrie épipolaire du capteur stéréo ayant servi à la prise de vues.

**Plusieurs passes** Nous pouvons imaginer un appariement en 3 passes :

1. établir des correspondances entre les points de deux lignes épipolaires conjuguées ; faire ceci pour tous les couples d'épipolaires conjuguées couvrant les deux images. Tous ces couples sont traités indépendamment les uns des autres.

2. recalculer ces appariements, en tenant compte cette fois de contraintes globales (c.à.d. les résultats trouvés pour les lignes adjacentes à la ligne courante).

3. éventuellement lisser une dernière fois la carte de disparité obtenue, en utilisant là aussi des contraintes globales et des hypothèses sur les images (contraintes de régularité et de cohérence).

Nous traitons pour l'instant uniquement la première phase ; la seconde sera évoquée en 3.5.

## 3.2 Méthode

Nous disposons donc de deux lignes épipolaires conjuguées $L_1$ et $L_2$ dont nous voulons apparier les points. Ces lignes sont constituées de $N$ et $M$ pixels :

$$\begin{cases} L_1 & = & (P_1 P_2 & \ldots & P_N) \\ L_2 & = & (Q_1 Q_2 & \ldots & Q_M) \end{cases}$$

Pour les apparier de façon optimale, nous utilisons la programmation dynamique, méthode d'optimisation combinatoire nous permettant de trouver une suite de transitions de coût minimal pour aller d'un état initial à un état final.

**États** Les états sont notés $E_{ij}$. L'état $E_{ij}$, $i = 0 \ldots N$, $j = 0 \ldots M$ signifie : "nous avons traité les segments $(P_1 \ldots P_i)$ et $(Q_1 \ldots Q_j)$ des droites $L_1$ et $L_2$, et apparié leurs points de façon optimale".

L'état initial est $E_{00}$ : rien n'est fait.

L'état final est $E_{NM}$ : les deux droites sont entièrement traitées.

**Transitions** Plaçons-nous en un état $E_{ij}$ ; nous avons donc apparié de façon optimale les points $(P_1 \ldots P_i)$ avec les points $(Q_1 \ldots Q_j)$. De l'état $E_{ij}$, nous autorisons seulement 3 types de transitions.

`match` nous apparions le point $P_{i+1}$ avec le point $Q_{j+1}$.
Nous nous trouvons donc dans l'état $E_{i+1,j+1}$.

`saut1` nous sautons le point $P_i$, pour examiner le reste de la droite $L_1$.
Nous nous retrouvons dans l'état $E_{i+1,j}$.

`saut2` nous sautons le point $Q_j$, pour examiner le reste de la droite $L_2$.
Nous nous retrouvons dans l'état $E_{i,j+1}$.

Chacune de ces transitions a un certain coût, et la programmation dynamique nous permet de trouver l'enchaînement de transitions conduisant de l'état initial $E_{00}$ à l'état final $E_{NM}$ dont la somme des coûts est minimale. Il s'agit simplement d'une recherche de chemin optimal dans le graphe complet (sans cycle) constitué de toutes les transitions, affectées de leur coût.

C'est la détermination de ces coûts qui constitue toute la difficulté ; elle fait l'objet de la partie suivante.

## 3.3 Coûts de transition

Nous allons calculer des probabilités d'appariement (ou de non-appariement). La programmation dynamique minimisant une somme de coûts, nous prendrons en fait les logarithmes des probabilités (pour minimiser le produit).

**Probabilité d'une transition `match`** Nous sommes en $E_{ij}$. Soit $C$ une mesure de corrélation, et $c_0 = C(P_{i+1}, Q_{j+1})$. Notons $M(P_{i+1}, Q_{j+1})$ la fonction d'appariement, c.à.d. $M = 1$ si les points sont appariés, et $M = 0$ sinon. Alors :

$$Prob(M = 1|C = c_0) = \frac{Prob(C = c_0|M = 1).Prob(M = 1)}{Prob(C = c_0)}$$

▷ $Prob(M = 1)$ est la probabilité que les deux points $P_{i+1}$ et $Q_{j+1}$ se correspondent *a priori*, sans autre connaissance sur leur ressemblance (= la mesure de corrélation). On peut la fixer à $\frac{1}{\text{nombre de points possibles}}$ × probabilité de visibilité. La probabilité de visibilité est égale à (1 − le taux d'occultation) ; ce dernier est fixé pour toute l'image (par exemple 0.2 si 80% des points d'une image sont visibles dans l'autre).

▷ $Prob(C = c_0)$ est la probabilité que la mesure $C$, appliquée à deux points quelconques, renvoie la valeur $c_0$. Cette probabilité est estimée en mesurant la fonction de distribution de $C$ le long des deux épipolaires considérées : on construit un histogramme de toutes les valeurs que prend $C$ le long de ces droites.

▷ $Prob(C = c_0|M = 1)$ est la plus délicate à calculer, car pour disposer d'une estimation, il faudrait déjà connaître des points appariés... En pratique, nous apparions quelques points des droites sur un simple critère de corrélation, et comme précédemment, nous déterminons la fonction de distribution de $Prob(C = c_0|M = 1)$ en fonction de $c_0$.

**Probabilité d'une transition `saut1` ou `saut2`** Puisque $M$ ne peut valoir que 0 ou 1 :

$$Prob(M = 0|C = c_0) = 1 - Prob(M = 1|C = c_0)$$

Ceci nous donne la probabilité que $P_{i+1}$ et $Q_{j+1}$ ne se correspondent pas. Comme rien ne permet de déterminer si nous avons affaire à une transition `saut1` ou `saut2`, nous fixons leurs probabilités égales chacune à $\frac{1-Prob(M=1|C=c_0)}{2}$.

## 3.4 Résultats d'appariements

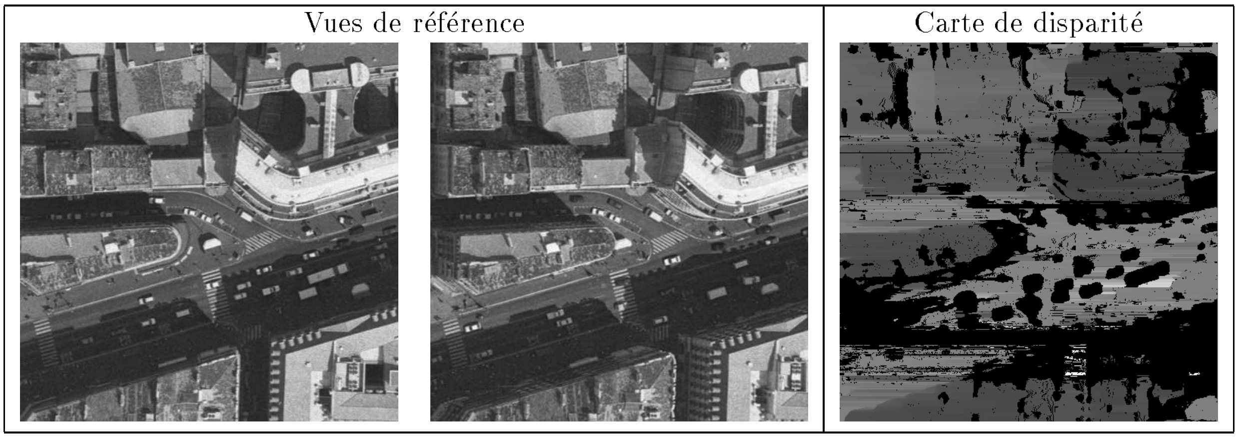
Nous montrons ici les résultats d'appariements denses pour deux couples d'images.

**Marseille** Voici deux images de référence, vues aériennes[2] de la ville de Marseille. Ces images sont rectifiées, donc les épipolaires sont horizontales, et les lignes de même ordonnée se correspondent. Le résultat de l'appariement est aussi montré ci-dessous, sous forme d'une carte de disparité. Le niveau de gris indique l'ordre de grandeur de la disparité en chaque point, donc son élévation. Les zones noires correspondent à des points non appariés.

Notons encore une fois qu'il n'y a aucune contrainte de cohérence d'une ligne à la suivante ; on remarque bien que les véhicules n'ont pas été appariés, car ils ne se retrouvent pas d'une image à l'autre (la scène n'est pas statique, et les voitures se sont déplacées entre les deux prises de vues). D'autres zones n'ont pas été appariées parce qu'elles étaient simplement occultées dans l'autre image (en particulier,

2. Propriété d'ISTAR

zone en forme de croissant au centre de l'image, au-dessus des voitures).




**Coupe de Fruits** Voici deux autres vues de référence, et la carte de disparité trouvée :


Ici, la scène est bien statique ; on voit que l'algorithme d'appariement détecte néanmoins de nombreuses occultations (sur les frontières de disparité essentiellement).

### 3.5 Développements

Notre modèle est simple, et ne prend pas en compte des contraintes de voisinage. Voici comment nous pourrions intégrer ces contraintes dans une deuxième passe.

Soit $D$ une fonction de disparité sur un voisinage des points $P_{i+1}$ et $Q_{j+1}$. Après notre première passe d'appariement, nous disposons d'une valeur $d_0 = D(P_{i+1}, Q_{j+1})$. Nous cherchons alors $Prob(M = 1 | (C = c_0) \cap (D = d_0))$. Or :

$$Prob(M = 1 | (C = c_0) \cap (D = d_0)) = \frac{Prob(C = c_0 | M = 1).Prob(M = 1 | D = d_0).Prob^2(M = 1)}{Prob(C = c_0).Prob^2(D = d_0)}$$

En partie droite de l'égalité nous restent à déterminer $Prob(M = 1 | D = d_0)$ et $Prob(D = d_0)$, que nous pourrons estimer à partir des images et des résultats de la première passe.

Cette deuxième passe est encore en cours de développement.

## 4 Résultats

Nous avons vu comment, à partir de deux vues de référence, nous pouvions établir des appariements, pour reconstruire ensuite un modèle presqu'euclidien point par point, puis synthétiser simplement d'autres vues de ce modèle.

Nous montrons ci-dessous quelques vues qu'il a été possible de synthétiser à partir des deux vues de référence de la coupe de fruits.

Vues synthétisées

## 5 Conclusion

Nous avons montré comment, étant données quelques vues d'une scène 3D, on pouvait synthétiser d'autres vues de cette même scène, pour par exemple simuler le déplacement d'une caméra virtuelle. Cela ouvre la voie à de nombreuses applications, dans tous les domaines où la réalité virtuelle peut être mise à profit.

Comme souvent en stéréovision, la partie la plus délicate est la mise en correspondance. Nous avons exposé ici un algorithme d'appariement dense entre deux images s'appuyant sur une optimisation de probabilités d'appariements. Le modèle est séduisant, mais demande encore à évoluer, en particulier pour la prise en compte de contraintes de cohérence globale.

## Références


[BM95] B. Boufama and R. Mohr. Epipole and fundamental matrix estimation using the virtual parallax property. In *Proceedings of the 5th International Conference on Computer Vision, Cambridge, Massachusetts, USA*, pages 1030–1036, June 1995.

[Fau92] O. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In G. Sandini, editor, *Proceedings of the 2nd European Conference on Computer Vision, Santa Margherita Ligure, Italy*, pages 563–578. Springer-Verlag, May 1992.

[GLY92] D. Geiger, B. Ladendorf, and A. Yuille. Occlusions and binocular stereo. In G. Sandini, editor, *Proceedings of the 2nd European Conference on Computer Vision, Santa Margherita Ligure, Italy*, pages 425–433. Springer Verlag, 1992.

[HS94] R. Hartley and P. Sturm. Triangulation. In *Proceedings of* ARPA *Image Understanding Workshop, Monterey, California*, pages 957–966, November 1994.

[LMR95] Z. D. Lan, R. Mohr, and P. Remagnino. Robust matching by partial correlation. In *Proceedings of the sixth British Machine Vision Conference, Birmingham, England*, pages 651–660, September 1995.

[Luo92] Q.T. Luong. *Matrice fondamentale et autocalibration en vision par ordinateur.* Thèse de doctorat, Université de Paris-Sud, Orsay, France, December 1992.

[OK85] Y. Ohta and T. Kanade. Stereo by intra and inter-scanline search using dynamic programming. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 7(2):139–154, 1985.