
Méthodologie de conception de motifs dirigée par la distance de Hamming et la géométrie épipolaire en lumière structurée

Xavier Maurice, Christophe Doignon

*Laboratoire des Sciences de l'Image, de l'Informatique et de la Télédétection
(UMR CNRS 7005) Université de Strasbourg*

Bd Sébastien Brant, BP 10413

F-67412 Illkirch cedex

xavyeah.maurice@gmail.com, c.doignon@unistra.fr

RÉSUMÉ. En vision artificielle par lumière structurée, la projection d'un motif sur un objet permet de capturer la forme de sa surface à un instant donné, en effectuant généralement plusieurs acquisitions. Cette technique peut cependant s'avérer inefficace lorsque l'objet est en mouvement. Il devient alors nécessaire de ne faire appel qu'à une seule prise d'images pour analyser la forme d'une région d'intérêt en mouvement, et des motifs intégrant des propriétés intrinsèques de corrections doivent alors être envisagés. Dans cet article nous présentons une nouvelle méthode d'appariement d'indices visuels composant des motifs de lumière structurée codée. L'originalité de la méthode consiste, à travers un codage spatial, à exploiter conjointement la géométrie épipolaire et une distance de Hamming désirée, critère dont la valeur minimale peut être spécifiée dès la conception, assurant un niveau minimal de redondance des informations au sein du motif à projeter. En contrepartie, quand la valeur de celle-ci est élevée, les algorithmes de codage/décodage ont à faire face à un nombre considérable de codes candidats. Dans un premier temps, nous montrons que la distance de Hamming moyenne est un critère statistique pertinent pour prédire les propriétés globales du futur motif, et nous fournissons un moyen de la calculer. Nous présentons un algorithme de codage basé sur la théorie des matrices parfaites, la distance de Hamming et le paradigme exploration/exploitation, où l'unicité des codes est examinée dans l'espace monodimensionnel des adresses des codes au lieu de celui du motif matriciel, comme c'est le cas dans la littérature. Dans un second temps, mettant en synergie la conformation des indices visuels et la géométrie épipolaire, nous montrons qu'il est possible de prédire la façon dont ces indices vont se projeter sur le plan image, permettant ainsi de simplifier la segmentation, le codage et le décodage. Les résultats montrent une réduction significative de la complexité algorithmique et la possibilité de fournir des motifs en temps réel, ou bien d'une taille bien supérieure à ce qui existe à ce jour. Finalement, des reconstructions de scènes diverses et effectuées aussi en temps réel, à partir d'une seule image des motifs projetés, sont reportées, pour illustrer le potentiel applicatif de notre approche.

ABSTRACT. Structured light-based patterns provide a way to capture the state of an object shape. However it may be inefficient when the object is freely moving or when its surface contains highly curved parts. To deal with unstructured and dynamical environments, only one shot is required for capturing the shape of a moving region-of-interest. Then robust patterns and real-time capabilities must be targeted. In this paper we present a new matching

technique for the pattern visual features, through the numerical coding of the associated symbols. The novelty of the proposed method is to use both the epipolar geometry and a desired Hamming distance of this stereovision-like system, the latter criterion can be specified by its minimal value for the pattern design. This ensures a high degree of redundancy within the overall pattern to project. The counterpart is the big amount of codes the coding/decoding algorithms have to face with a high desired Hamming distance. In the first part, we show that the mean Hamming distance is a useful criterion for characterizing the future pattern and we give a way to predict its value. Furthermore, to ensure the local uniqueness of every code with consideration of many incomplete ones, the Perfect Map theory is involved. Then, we describe a pseudorandom/exhaustive algorithm to build very large patterns in a very short time, thanks to a splitting strategy which performs the Hamming tests in the 1-D code address space instead of the 2-D pattern array. This leads to a significant reduction of the computational complexity and it may be applied to other purposes. Second, the conjunction of features conformation and epipolar geometry simplifies the segmentation as it provides a better understanding of the projected features in the image. The results show that epipolar-aligned patterns can be quickly produced thanks to this coding scheme which yields far more redundant and/or bigger coding maps than what have been achieved in the literature. Finally, real-time surface reconstructions from single images are reported and results are compared to the best known ones which are outperformed in many cases.

MOTS-CLÉS : lumière structurée codée, matrices parfaites, géométrie épipolaire, reconstruction de surfaces, distance de Hamming minimum.

KEYWORDS: coded structured light, perfect maps, epipolar geometry, surfaces reconstruction, minimum Hamming distance.

DOI:10.3166/TS.29.229-254 © 2012 Lavoisier

Extended abstract

Coded structured light is a useful technique for the 3-D reconstruction of poorly textured scene areas. The codes, uniquely associated with visual features of the projected pattern, are used to solve the correspondence problem using local information only, with robustness property against disturbances like high curvatures, occlusions, out of field-of-view, out-of-focus. Real-time reconstruction is even possible with pseudo-random arrays, where the encoding is done in a single pattern using spatial neighborhood. Keeping a high Hamming distance between all the codes allows to correcting mislabeled features and thus it ensures patterns to be robust to noise.

Up to now, the proposed coding schemes ensured a desired Hamming distance between all the codes/features of the pattern which is generated off-line, beforehand, producing Perfect Sub Maps (PSM). But knowing the epipolar geometry of the projector-camera system, one has to ensure the Hamming distance only between features that are along the same epipolar line, because these only can produce correspondence ambiguity during the decoding process. As for such a new coding scheme, the Hamming distance must be checked only within subsets of pattern

elements, the patterns are globally far less constrained and therefore some of them can be generated at a video frame rate (depending on the desired dimensions of the pattern, the number of symbols and the desired Hamming distance). This type of pattern matrix is named Sub Perfect Sub Maps (SPSM), due to the local uniqueness.

In this paper, we present a new pattern features matching technique for real-time reconstructions through the numerical coding of the associated symbols. The novelty of the proposed method is in the use of both the epipolar geometry and a desired Hamming distance, specified by its minimal value for the pattern design. This ensures a high degree of redundancy but the counterpart is the big amount of codes the coding/decoding algorithms have to face with a high desired Hamming distance.

The epipolar line homography

Consider a 2-D point in one of the image planes (projector or camera) of a stereo system. It is known that the corresponding epipolar line in the other image plane can be computed thanks to the fundamental matrix F . This property reduces the 2-D matching problem to a one-dimensional search problem. Although this positioning constraint is commonly used to match features, it is not the case with the local orientation related to the feature's shape. Indeed, in the general case, if the scene to reconstruct is *a priori* unknown, it is not possible to predict how a straight line projected onto the scene and issued from the projector behaves on the image plane of the camera (IPC), as it usually depends on the scene. Nevertheless, one can notice a singularity: in the image plane of the projector (IPP), if a pencil of lines \mathbf{P}_p intersects exactly at the epipole \mathbf{e}_p , then the corresponding pencil of lines in the IPC, \mathbf{P}_c , is a set of epipolar lines, thus passing through the epipole \mathbf{e}_c . This property is due to the one-dimensional homography which relates the two sets of corresponding epipolar lines, independently to the scene structure:

$$\mathbf{P}_p \equiv F [\mathbf{e}_c]_{\times} \mathbf{P}_c .$$

Then, features oriented along epipolar lines in the IPP will project along the corresponding epipolar lines on the IPC whatever the scene is. This avoids looking for any features locations and directions and provides rotational invariance.

Robust Grid-based Pattern coding

Different approaches have been recently summed up for dense and sparse pattern coding, in the field of structured light (see for instance (Salvi, 2004)). For our application of concern, real-time 3-D reconstruction of moving surfaces, we look for approaches based on spatial neighborhoods. Indeed, time-multiplexing approaches do not allow reconstruction of moving scenes and direct coding is too sensitive to illumination changes. Finally using different colors, shapes and intensity levels in the close neighborhood of interest features lead to a huge number of neighborhood combinations which increases their discriminating power, yielding patterns robust to noise, occlusion, out-of-focus, or highly curved surfaces. In recent works, adaptive structured light patterns have appeared (Koninckx, 2006; Adan, 2009). The pattern is modified in real-time to adapt it to local properties of the scene, mainly by local

features density adaptation when high curvatures or thin objects are detected. Local constraints on the symbols number can also be included to better handle particular surface reflectance properties through the use of locally adapted colors. Actually, these adaptive patterns have been proposed without a coding framework, as usually this kind of re-coding is time consuming and carried out off line. Moreover, throughout this article, we propose a new class of coded patterns where coding constraints depend on the epipolar geometry of the whole stereo-like system. If the projector-camera transformation is not constant over time, then such patterns have to be recorded in real-time. For these reasons, a real-time coding algorithm would open new perspectives.

A matrix $P = (m \times n; u \times v; C; H_{Min})$ is a Perfect Map (PM) if it is a $(m \times n)$ array in which all the C -ary $(u \times v)$ sub-matrices appear exactly once. A Perfect Sub Map (PSM) has the same properties as PM except that not all the sub-matrices are involved. The maps are generated with algorithms based on brute-force methods (trials and errors) and are suited to enforce a high Hamming distance ($H_{Min} > 1$).

The objective of this work is to perform such coding, embedding high data redundancy, thanks to the Perfect Maps theory and with the Hamming distance as the robustness criterion. For the same neighborhood size, such algorithms aim at:

1. Maximizing the minimum Hamming distance to increase the corrections capabilities within the pattern.
2. Minimizing the number of symbols to reduce the risks of incorrect segmentations and classification of features.
3. Maximizing the pattern size to allow the precise reconstruction of small objects, better handle occlusions and surface curvature variations.

The well-known bottleneck of the existing brute-force algorithms for pattern generation is the test of uniqueness, also known as *H Tests*, that is the set of tests to be performed for any candidate code with respect to the already included ones. We show that it is possible to achieve large patterns with high correction capabilities very quickly if the *H Tests* are performed in the codes index space instead of the 2-D space array (the pattern matrix space). First, we address the simple case for $H_{Min} = 1$, then for higher values. In the latter case, it is possible to divide the search, by splitting the Hamming distance values via one or several intermediate codes (Maurice, 2012). This has a significant effect on the computational complexity as, in practice, very big patterns can be built in a very short time.

Epipolar-based pattern coding

When the epipolar geometry is known, Sub Perfect Sub Maps (SPSM) can be carried out. They have the same properties as PSM except that the tests of uniqueness (*H Tests*) are performed only between subsets of codes, thus corresponding to a close neighborhood. Herein, the subsets are placed on the epipolar lines. The proposed epipolar-based pattern algorithm computes such SPSM. The results show that epipolar-aligned patterns can be quickly produced thanks to this coding scheme which yields far more redundant and/or bigger coding maps than

what has been achieved in the literature. This scheme speeds up both the decoding and segmentation processing steps, so as to reach the video real-time achievement. Finally, real-time reconstructions from single images are reported and results are compared to the best known ones which are outperformed in many cases.

We have proposed two new contributions towards the real-time coding of structured light patterns based on the spatial neighborhood coding approach. The first one brings the concept of a sizeable correcting window to find the best balance between exploration and exploitation of the solution space and is independent of the epipolar geometry, yielding perfect sub maps. The second contribution exploits the epipolar geometry to decrease the constraints complexity of the uniqueness test, leading to the generation of sub perfect sub maps (SPSM). Both contributions provide a significant improvement w.r.t. the state of the art and the latter allows the real-time generation of patterns of very large size while providing correction capabilities with only few symbols.

1. Introduction

La projection de lumière structurée est couramment utilisée pour réaliser des reconstructions 3D de scènes *a priori* inconnues, faiblement ou non texturées, là où les systèmes stéréoscopiques classiques sont limités. Elle est fondée sur l'apport de signaux prédéfinis injectés dans une scène inconnue. En capturant des images de l'ensemble, il est possible d'estimer la profondeur, voire de reconstruire localement des surfaces peu texturées. Le problème principal auquel on est confronté avec des données éparses (indices visuels) est la mise en correspondance entre les indices du motif projeté et leur correspondant dans l'image acquise. Une des solutions pour résoudre ce problème peut être d'associer un code à chaque indice du motif. Pour ce faire, différentes stratégies ont déjà été proposées (voir Salvi, 2004).

Par le passé, la vision par projection de lumière structurée de lignes, d'indices colorés ou de plusieurs niveaux d'intensité, a été largement utilisée pour capturer des surfaces d'objets statiques ou en mouvement contrôlé pour une reconstruction dense ou à l'aide d'indices visuels (Morita, 1988 ; Griffin, 1992 ; Salvi, 1998 ; Zhang, 2002 ; Koninckx, 2006 ; Chen, 2007).

Dans le cas d'objets se déplaçant librement, Weise (Weise *et al.*, 2007) à l'aide de la stéréovision et du décalage de phase, et Young (Young *et al.*, 2007) avec la projection d'un réseau de droites, ont proposé des techniques de reconstruction de surfaces à partir d'une seule acquisition d'image. Kawasaki *et al.* (2008) réalisent une reconstruction dense des surfaces grâce à l'utilisation des informations d'adjacence locale au sein d'une grille de motifs. Récemment, Sagawa *et al.* (2009) ont présenté une méthode de reconstruction 3-D rapide à l'aide de la propagation de croyance et d'un codage par séquences de De Bruijn sur des droites (De Bruijn, 1946). Toutes ces techniques nécessitent d'associer un indice visuel (ou primitive) à un symbole numérique puis d'effectuer l'appariement des indices visuels capturés – et déformés – avec ceux qui composent le motif original avant sa projection.

Pour le contexte applicatif qui nous concerne, à savoir la reconstruction de surfaces en mouvement, nous avons développé une technique de reconstruction en temps réel qui fait appel à la projection de lumière structurée. A cette fin, nous avons retenu les approches appartenant à la famille des codages par voisinages spatiaux. En effet, celles relevant du multiplexage temporel ne sont pas adaptées à la reconstruction de scènes dynamiques (Guhring, 2001 ; Ishii *et al.*, 2007 ; Pribanic *et al.*, 2009) et les méthodes employant le codage direct (Hung, 1993) s'avèrent beaucoup trop sensibles aux changements de luminosité ou aux perturbations spéculaires pour être utilisées de manière fiable dans ce contexte. De plus, le codage par voisinage spatial peut être mis en œuvre en s'appuyant sur une distance de Hamming élevée. Cette distance mesure l'écart entre les voisinages spatiaux dans le motif ; plus elle est élevée et plus il sera facile d'identifier ce code dans l'image du motif. Une distance de Hamming élevée offre la possibilité de corriger des indices visuels mal classifiés à partir d'informations locales (dès qu'elle est supérieure à 1, valeur limite qui assure l'unicité du voisinage – ou code – dans le motif).

Dans la catégorie des techniques par voisinage spatial, un code identifie le voisinage (une sous-matrice le plus souvent) par l'agencement de C symboles. Cet agencement est le résultat d'une stratégie de codage qui identifie sans ambiguïté le code dans tout le motif. Ainsi, pour décoder la partie visible d'un motif capturé, plusieurs approches ont été proposées et la plupart d'entre elles sont résumées dans (Batlle, 1998 ; Salvi, 2004). Les approches par voisinage spatial sont plus appropriées pour distinguer les indices visuels associés aux symboles, sur une seule prise d'images. Que cette association se fasse sur des couleurs, des formes géométriques ou des niveaux de gris (figure 1), on obtient avec ce type de codage un nombre considérable de combinaisons de voisinages, donc de codes, ce qui a pour effet d'élever le potentiel de discrimination entre les indices visuels, et finalement de générer des motifs robustes au bruit, au flou, et aux surfaces présentant de fortes courbures (Adan, 2009 ; Maurice *et al.*, 2010).

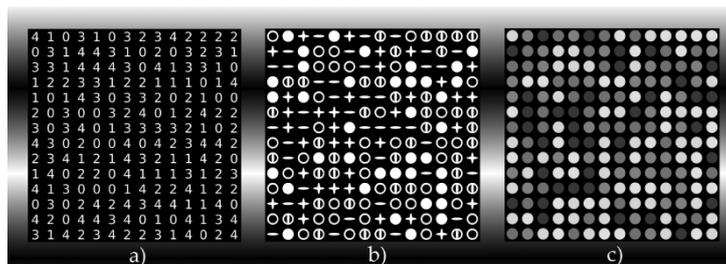


Figure 1. Exemples de matrices sous-parfaites 5-aires, de dimensions (14×14) et possédant une distance de Hamming de $H = 5$ entre tous les voisinages (3×3) , et représentées par a) des nombres ; b) des formes géométriques ; c) des niveaux de gris. Au moins deux erreurs par voisinage peuvent être corrigées

1.1. Motivations et contributions

La précision d'une reconstruction 3-D obtenue par projection de lumière structurée est liée entre autres au codage des motifs. C'est très significatif pour des surfaces complexes. Pour étayer cela, on observe que les travaux dans ce domaine se concentrent sur le codage en vue d'améliorer la précision des reconstructions. Cet article se situe dans cette optique et l'objectif que nous avons poursuivi pour satisfaire les contraintes applicatives est d'élaborer un schéma de décodage rapide pour réaliser des reconstructions en temps réel. Les motifs générés embarquent donc une forte redondance des informations, représentée par la distance de Hamming.

Dans cet article, nous montrons qu'en exploitant judicieusement la distance de Hamming et la géométrie épipolaire, de nouveaux motifs plus efficaces pour l'appariement robuste sur des scènes complexes peuvent être construits. La connaissance de la géométrie épipolaire, qui permet de restreindre les possibilités de correspondance entre les indices d'un couple d'images, est ici mise à profit dès la conception du motif afin de retrouver la position et l'orientation des indices capturés. Nous montrons que la géométrie épipolaire permet de savoir comment des indices possédant une information directionnelle contrôlée dans le plan image du projecteur (PIP), peuvent se comporter dans le plan image de la caméra (PIC). En d'autres termes, si la géométrie des indices visuels est correctement choisie, il est possible de générer un motif en positionnant et en orientant ceux-ci dans le PIP de manière à obtenir une projection souhaitée, quelle que soit la nature de la scène.

1.2. Travaux antérieurs

En préambule, nous rappelons que la distance de Hamming, H , entre deux mots de code (ou simplement code, par la suite) est le nombre de positions où les symboles diffèrent. Ainsi, une matrice C -aire, $P = (m, n; u, v)$, de dimensions $(m \times n)$ est une matrice parfaite (*Perfect Map* - PM) si toutes les sous-matrices $(u \times v)$ apparaissent une seule fois. Donc, si chaque sous-matrice $(u \times v)$ représente un code, la distance de Hamming entre toutes les sous-matrices d'une matrice parfaite est au moins égale à 1. Une matrice parfaite est l'équivalent 2-D des séquences de De Bruijn (De Bruijn, 1946) qui ont été très largement employées en lumière structurée (Patterson, 1994 ; Mitchell, 1995 ; Salvi, 1998 ; Pagès, 2005).

Etant donné une taille de voisinage $W = (u \times v)$, la construction d'une matrice parfaite aux dimensions $(m \times n)$ souhaitées pour le motif dans le but d'obtenir une reconstruction précise n'est pas une tâche triviale. Afin d'atteindre des valeurs élevées de m et de n , Pagès (2005) a utilisé un codage hybride basé sur les séquences de De Bruijn et mis en œuvre avec des couleurs et des niveaux de gris.

A notre connaissance, les premières constructions directes de motifs basés sur les matrices parfaites ont été proposées par (Etzion *et al.*, 1988 ; Hulbert *et al.*, 1993), dans le cas binaire ($C = 2$). Les algorithmes proposés ne permettaient cependant pas de choisir la taille du motif, ni d'imposer une distance de Hamming supérieure à 1. C'est sur ces deux paramètres que les algorithmes « brute-force » se sont montrés

plus performants que les méthodes directes. De tels algorithmes itératifs remplissent un motif matriciel C -aire d'une taille désirée en menant un ensemble de tests-erreurs successifs. A chaque itération, la distance de Hamming souhaitée est testée entre un nouveau mot de code candidat et tous ceux déjà intégrés dans le motif.

Morano (1998) et plus récemment Albitar *et al.* (2007) à l'aide d'indices visuels géométriques puis Claes (2008) avec des spots colorés ont proposé des constructions de motifs basées sur les algorithmes « brute-force ». Au moyen de ce type d'algorithme, comme toutes les sous-matrices ne sont pas présentes dans le motif, les matrices obtenues sont appelées sous-parfaites (*Perfect Submaps* - PSM) et récemment Adan *et al.* (2009) ont employé les PSM pour réaliser des motifs colorés hexagonaux. Les limites de cette famille d'algorithmes concernent le nombre de tests à effectuer sur la distance de Hamming entre chaque couple de codes. La complexité exponentielle caractérisant cette approche, la construction de motifs peut nécessiter plusieurs heures/jours de calculs pour des motifs de taille suffisante pour une utilisation pratique. Les dimensions des motifs étant donc limitées, il en résulte une précision de reconstruction limitée également.

2. Statistiques pour la conception de motifs

Dans cette section, nous présentons en premier lieu les paramètres qui agissent sur les statistiques des motifs ($W = u \times v, C, H_{Min}$), avant d'aborder à la section suivante la génération de motif proprement dite.

2.1. Distance de Hamming minimum

La distance de Hamming $H(C_i, C_j)$ entre les codes C_i et C_j , est égale au nombre de positions r dans les deux codes où des symboles diffèrent :

$$H(C_i, C_j) = \sum_{r=1}^W \delta_r \quad ; \quad \delta_r = \begin{cases} 0 & \text{si } C_{ir} = C_{jr} \\ 1, & \text{sinon.} \end{cases} \quad (1)$$

La distance de Hamming minimum H_{Min} d'un ensemble de codes est la plus petite distance de Hamming entre tous les couples de mots de code possibles dans cet ensemble. Il est bien connu en théorie de l'information (Peterson, 1972), que pour corriger q symboles mal identifiés, il faut $H_{Min} \geq 2q + 1$, d'où l'intérêt de d'assurer une valeur de H_{Min} la plus élevée possible lors du codage des motifs.

2.2. Distance de Hamming moyenne

Nous reconsidérons l'idée largement répandue qu'utiliser moins de symboles fournit toujours la meilleure solution. En effet, considérer plus de symboles dans le motif peut augmenter les capacités de correction de celui-ci, mais ceci est de moins en moins significatif au fur et à mesure que le nombre de symboles augmente. Pour illustrer ceci, considérons les distributions typiques du nombre de tests d'unicité basée sur la distance de Hamming entre chaque couple de codes (figure 2). Pour

plusieurs valeurs du nombre de symboles C , pour des voisinage (3×3) et pour $H_{Min} = 1$, on peut observer que quand la valeur de C augmente, le nombre de couples de codes qui ont une faible distance de Hamming, H , diminue. Cela signifie qu'il est plus probable de décoder tous les couples présentant une faible valeur de H et un nombre élevé de symboles qu'avec un faible nombre de symboles. Si ces couples de codes ont pu être décodés, alors tous les autres couples présentant une distance de Hamming supérieure à H peuvent potentiellement être employés pour corriger d'éventuelles erreurs, en dépit du fait que la distance de Hamming minimum désirée, H_{Min} , pour le motif ait été choisie égale à 1 seulement. Cela signifie donc qu'un nombre de symboles porté à $C = 8$ par exemple autorise plus de corrections dans le motif que si $C = 3$. Cependant, on constate aussi que plus C augmente, moins le gain (en capacités de corrections) entre deux valeurs successives de C est important (diminution de l'écart entre les courbes). En conclusion, l'utilisation de plus de symboles (jusqu'à une valeur qui n'induit pas de problème de discrimination des indices visuels associés) augmente le potentiel correctif embarqué dans le motif.

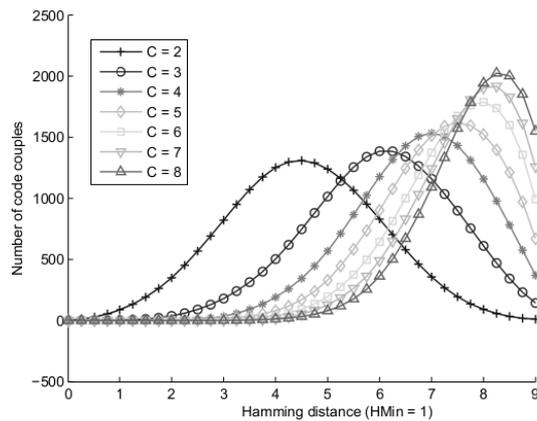


Figure 2. Distributions des distances de Hamming de motifs conçus avec C symboles, $H_{Min} = 1$ et $W = 9$ (moyenne issue de 100 générations de matrices sous-parfaites, de tailles identiques, pour différentes valeurs de C)

Il ressort de ces premiers résultats qu'il serait intéressant d'étudier l'évolution des distributions de la distance de Hamming et, par comparaison avec celles obtenues *a posteriori* sur des motifs générés de la figure 2, de prédire le potentiel correctif que présentera le futur motif en fonction de plusieurs paramètres d'entrée. Pour conduire cette étude, nous supposons que l'évolution de la distance de Hamming moyenne représente suffisamment finement ce potentiel correctif à travers le comportement de chaque distribution (bien que l'écart type évolue lui aussi si on s'en tient uniquement aux statistiques du premier et du deuxième ordre).

2.2.1. Définition de la distance de Hamming moyenne

Etant donné un ensemble S de N codes C_i de longueur W , la distance de Hamming moyenne, $\bar{H}(S)$, est définie par la somme des distances de Hamming $H(C_i, C_j)$ entre tous les couples de codes de S , divisée par le nombre total de couples

$$\bar{H}(S) = \frac{\sum_{i=1}^N \sum_{j=i+1}^N H(C_i, C_j)}{N(N-1)/2} \quad (2)$$

2.2.2. Estimation de \bar{H} a priori

Lors de l'élaboration d'un codage basé sur les matrices sous-parfaites, il n'y a pas lieu de favoriser un symbole plutôt qu'un autre. Par conséquent, les algorithmes qui supposent que les symboles soient uniformément distribués peuvent être caractérisés à travers une grandeur statistique telle que la distance de Hamming moyenne \bar{H} .

Proposition. En supposant une distribution uniforme des symboles dans des codes de longueur W et choisis dans un alphabet C -aire, la distance de Hamming moyenne \bar{H} entre tous les couples de codes est égale à

$$\bar{H} = W - W/C . \quad (3)$$

Preuve. Soit $A = \{A_1, \dots, A_W\}$ et $B = \{B_1, \dots, B_W\}$ deux codes de longueur W , avec A_i et $B_i \in \{0, 1, \dots, C-1\}$. Les symboles A_i et B_i peuvent prendre toutes les valeurs parmi C avec une équiprobabilité de $1/C$. Il s'ensuit que la probabilité que le même symbole apparaisse dans A et dans B à la $i^{\text{ème}}$ position correspond à

$$\begin{aligned} p(A_i = B_i) &= p(A_i = 0) \times p(B_i = 0) + \dots + p(A_i = C-1) \times p(B_i = C-1) \\ &= \sum_{j=0}^{C-1} p(A_i = C_j) \times p(B_i = C_j) = \sum_{j=0}^{C-1} \left(\frac{1}{C}\right) \times \left(\frac{1}{C}\right) = 1/C. \end{aligned}$$

L'espérance mathématique E du nombre de symboles identiques entre les codes A et B s'exprime par $E = p(A_1 = B_1) + p(A_2 = B_2) + \dots + p(A_W = B_W) = W/C$.

Finalement, par complémentarité, on obtient $\bar{H} = W - E$. ■

Sur la figure 3, on observe que les prédictions des valeurs de \bar{H} , pour plusieurs valeurs de H_{Min} , sont très proches de celles calculées *a posteriori* avec (1) à partir d'un grand nombre de motifs générés. Seul le cas $(C, H_{Min}) = (2, 5)$ présente un biais, mais ce cas est peu rencontré en pratique. Il ressort de ces résultats que l'évolution de \bar{H} vis-à-vis du nombre de symboles est très similaire à celle du potentiel correctif évoqué auparavant, la courbe s'infléchissant de plus en plus au fur et à mesure que le nombre de symboles augmente.

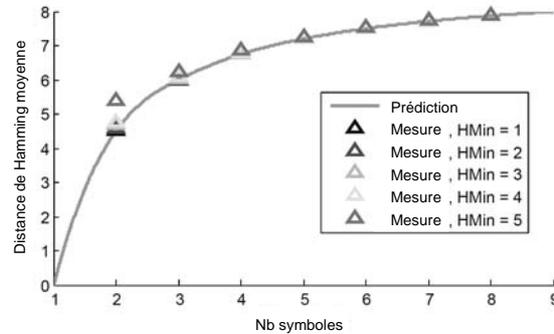


Figure 3. Valeurs de \bar{H} prédites à partir de (3) (courbe continue) et valeurs estimées a posteriori avec (2) (triangles) à l'aide de motifs générés (moyenne sur 100 motifs), pour 5 valeurs de H_{Min} et en fonction du nombre C de symboles ($W = 9$)

3. Génération de motifs matriciels

Dans cette section, nous présentons, en plusieurs étapes, la génération de motifs. En premier lieu, un nouvel algorithme de conception de motifs, de type « brute-force » est présenté. Bien que la taille des voisinages puisse être variable, la description ci-après considère des voisinages (3×3) , c'est-à-dire $W = 9$, ce qui permettra de comparer les performances obtenues avec d'autres travaux. Lors du processus de construction de motifs, les approches « brute-force » exécutent des tests pour valider l'intégration de nouveaux éléments, en se basant sur la valeur désirée de H_{Min} . La complexité algorithmique liée à ces tests est élevée. En effet, considérons que N codes aient déjà été incorporés, chaque mot de code qui peut être créé avec un nouveau code candidat doit être comparé avec les N codes déjà présents pour assurer que la distance de Hamming minimum désirée est respectée. Par conséquent, la complexité algorithmique liée aux tests pour chaque code candidat est $O(N)$. Nous proposons dans la section 3.2 une technique pour la réduire à 1 quand $H_{Min} = 1$ et dans la section 3.3 nous généralisons ceci pour des valeurs de H_{Min} plus élevées.

3.1. Méthodes existantes (matrices sous-parfaites)

Nous rappelons ici trois algorithmes de conception de motifs, dus à (Morano, 1998) (**M**), à (Claes, 2008) (**C**) et à (Maurice, 2010) (**X**), pour appréhender plus facilement par la suite ce que nous proposons.

Dans (**M**), des symboles choisis de manière aléatoire sont ajoutés un à un dans le motif. Le motif est rempli ligne par ligne, à partir du coin supérieur gauche. Si le code candidat contribue à obtenir une nouvelle sous-matrice (3×3) , alors la contrainte sur H_{Min} est vérifiée par rapport aux sous-matrices déjà présentes. Si c'est le cas, ce code est inclus puis un nouvel élément est traité, sinon les $C - 1$ autres

symboles de l'alphabet sont testés à leur tour. Si pour aucun d'entre eux la valeur de H_{Min} n'est vérifiée, le motif incomplet est abandonné et le processus recommence.

Dans (C), une recherche exhaustive est effectuée. Tous les éléments du motif sont d'abord initialisés à zéro. A chaque étape, le symbole 0 est testé en premier. Si avec le nouveau code qu'il forme et les N codes présents, la valeur de H_{Min} n'est pas respectée, tous les symboles des codes précédemment trouvés sont concaténés selon leur ordre d'insertion pour former un grand nombre en base C . Ce nombre est alors incrémenté et à chaque incrémentation, les voisinages correspondant aux chiffres incrémentés sont testés. Ainsi toutes les combinaisons possibles sont examinées, du nombre composé uniquement de 0 à celui composé uniquement de la valeur $C - 1$, en n'explorant cependant que les branches prometteuses de l'arbre des possibilités. Mais dès que la taille de la matrice devient conséquente et le nombre de symboles important, l'explosion combinatoire fait que très peu de possibilités peuvent être explorées en un temps raisonnable. De plus, les chiffres de poids forts étant très peu souvent incrémentés, la valeur 0 s'y trouve surreprésentée. La distribution des symboles n'est donc pas homogène.

3.2. Génération de motifs matriciels avec fenêtre de correction

Le paradigme exploration/exploitation, bien connu en intelligence artificielle, est appliqué dans (X). Grâce à l'apport d'une fenêtre de correction E (figure 4), il bénéficie des comportements aléatoire de (M) et exhaustif de (C). Ainsi, plus la fenêtre de correction E est grande, plus le comportement de l'algorithme sera (localement) exhaustif, une plus grande zone sera explorée pour corriger les éléments conflictuels. A l'opposé, plus la taille de la fenêtre de correction E est petite, plus le comportement sera aléatoire.

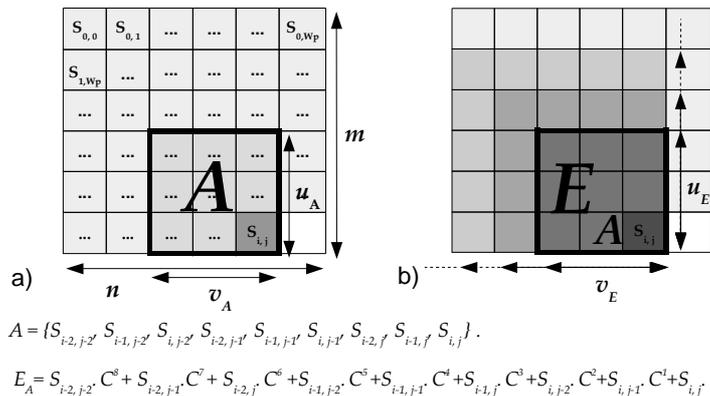


Figure 4. Algorithme brute-force avec fenêtre de correction E pour la génération de motifs basés sur les matrices sous-parfaites. a) L'ajout d'un nouvel élément dans la matrice crée un nouveau voisinage. b) Création de la fenêtre de correction E

Algorithme 1. *Génération de motifs matriciels avec fenêtre de correction*

Soit $(m \times n)$ les dimensions du motif désiré et $(u \times v)$ celles des voisinages (figure 4a). Les éléments sont traités ligne par ligne puis colonne par colonne.

1. Une matrice de taille $((m + u - 1) \times (n + v - 1))$ est créée, les $(u - 1)$ premières lignes et $(v - 1)$ premières colonnes sont remplies avec la valeur -1 pour éviter un traitement spécial des bords du motif.
 2. Chaque nouvel élément $S_{i,j}$ de la matrice est choisi aléatoirement, comme dans **(M)**. Cela permet, contrairement à **(C)** de maximiser la variance initiale et donc la distance de Hamming moyenne.
 3. Chaque nouvel élément $S_{i,j}$ $i \geq u, j \geq v$ engendre un nouveau mot de code A de voisinage $W_A = u_A \times v_A$. La distance de Hamming de A avec tous les autres mots de codes précédemment utilisés est testée pour vérifier la valeur de H_{Min} dans le cas d'une matrice sous-parfaite. (Dans le cas d'une sous-matrice sous-parfaite, sont testés seulement les codes appartenant au même sous-ensemble que A).
 4.
 - a. Si le test sur H_{Min} est concluant, le symbole $S_{i,j}$ est associé à cette cellule de la matrice et l'élément suivant est traité.
 - b. Sinon, une fenêtre de correction E de taille $u_E \times v_E$ est créée et les symboles s'y trouvant à l'intérieur sont convertis en un nombre en base C (C -aire), noté E_A (figure 4b). Les chiffres de E_A sont incrémentés en évoluant de leur valeur initiale $I_{i,j}$ à $I_{i,j} + C - 1$ (modulo C). De cette manière les C valeurs possibles sont examinées avant que le chiffre de poids supérieur ne soit à son tour incrémenté. Cela permet d'effectuer une recherche localement exhaustive seulement dans E . Le nombre d'incrémentations de chaque chiffre est stocké dans un compteur propre.
 - E_A est incrémenté tant que les modifications induites ne permettent pas d'obtenir un test sur H_{Min} concluant ou tant que tous les chiffres de E_A n'ont pas été incrémentés C fois.
 - Si E_A ne peut plus être incrémenté (toutes les valeurs ayant été testées), la matrice est effacée complètement et la recherche recommence, comme dans **(M)**.
-

Cet algorithme peut être vu comme un compromis entre l'approche **(M)**, purement aléatoire et **(C)**, purement exhaustif. Un équilibre entre exploration et exploitation est alors à trouver en fonction des ressources de calculs disponibles et du temps alloué au codage du motif. Un exemple de résultat est donné à la figure 5.

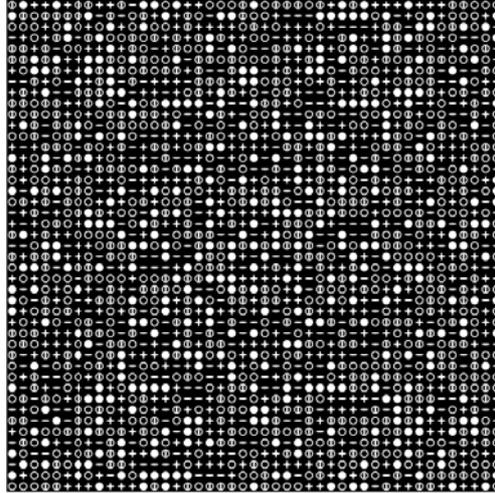


Figure 5. Exemple de génération d'une matrice sous-parfaite de taille 45^2 avec $C = 5$ symboles, des voisinages (3×3) et une fenêtre (5×5) de correction E

3.3. Optimisations proposées

Une tâche commune à toutes ces approches concerne les tests relatifs à la distance de Hamming. Quel que soit le comportement de l'algorithme brute-force (aléatoire, exhaustif, combinaison des deux avec fenêtre de correction), et quels que soient les ensembles de codes sur lesquels la distance de Hamming minimum est définie, la distance de Hamming entre un nouveau code et tous ceux déjà présents dans le motif doit être évaluée (H tests). A notre connaissance, ce test a toujours été réalisé dans l'espace 2-D du motif. Ainsi, quand une matrice pseudo-aléatoire ou matrice sous-parfaite de dimensions $(m \times n)$ est à construire, le nombre de H tests qui doivent être réalisés pour chaque nouveau code est au moins égal à $(m \times n)/2$. Par conséquent, la manière dont cette évaluation est conçue conditionne les performances des algorithmes de génération de motifs. L'algorithme décrit ci-après propose une réduction du nombre de H tests et reprend la structure générale de celle de l'algorithme (**X**) vu précédemment.

Auparavant, nous allons rappeler le théorème suivant issu de la théorie de l'information : le nombre de symboles requis pour créer N codes d'une longueur W avec une distance de Hamming minimum H_{Min} est égal à (Singleton, 1964) :

$$C \geq N^{1/(W-(H_{Min}-1))}.$$

Il s'ensuit que pour une valeur de H_{Min} fixée, l'espace des codes C -aire et de longueur W , noté $Card(\Omega_{C,W})$, est tel que

$$Card(\Omega_{C,W}) \leq C^{(W-(H_{Min}-1))}. \quad (4)$$

3.3.1. Réduction de la complexité pour $H_{Min} = 1$

Chaque code A ne devant apparaître qu'une fois au plus dans le motif, une variable booléenne, notée $\mathcal{B}(A)$, est associée à chacun d'entre eux pour indiquer sa présence dans la matrice. Pour contenir l'espace des codes, $Card(\Omega_{C,W})$, une table des C^W variables booléennes est construite et l'adresse (ou l'index dans la table) de chaque variable est associée à un seul code de $Card(\Omega_{C,W})$. Précisément, les W éléments correspondent aux coefficients A_i d'un polynôme en base C :

$$Adresse[\mathcal{B}(A)] = \sum_{i=1}^W A_i C^{W-i}. \quad (5)$$

Si $\mathcal{B}(A) = \text{Faux}$, alors le code A n'a pas déjà été inclus. Initialement, les variables booléennes sont toutes affectées avec la valeur *Faux*. Ainsi, pour un code A candidat, les variables booléennes pour lesquelles les codes possèdent une distance de Hamming $h \leq H_{Min}$ avec le code A sont testées.

Si toutes les variables booléennes ainsi testées sont égales à *Faux*, le H test est positif et le code A peut être inclus dans la matrice, sa variable booléenne $\mathcal{B}(A)$ est alors affectée avec la valeur *Vrai*.

Pour $H_{Min} = 1$, le seul code qui vérifie $h < H_{Min}$ est lui-même ($h = 0$). La complexité est donc en $O(1)$, autrement dit c'est un accès direct, et il est à comparer aux $m \times n/2 = 5000$ H tests quand ceux-ci sont réalisés dans l'espace de la matrice (motif de taille 100^2). Comme les variables booléennes associées aux codes doivent être sauvegardées au préalable, C^W bits d'espace mémoire sont nécessaires, ce qui, dans le cadre de notre étude, correspond à 8^9 bits = 16 Mo avec 8 symboles.

3.3.2. Généralisation à d'autres valeurs de H_{Min}

Etant donné C , W et H_{Min} , le nombre N_B de variables booléennes à examiner pour réaliser le H test dans l'espace des adresses des codes est (Maurice *et al.*, 2012) :

$$N_B(H_{Min}) = 1 + \sum_{h=1}^{H_{Min}-1} \frac{W!}{h!(W-h)!} (C-1)^h \quad (6)$$

Ceci est illustré sur la figure 6 pour $C = W = 3$ et $H_{Min} = 2$ (dernière ligne). Dans cet exemple, le code à tester est $A = \{2,2,2\}$ d'adresse 26. Le nombre de tests à effectuer est ici de $N_B(2) = 1 + \frac{3!}{(3-1)!} (3-1)^1 = 1 + 6 = 7$ pour assurer une différence de $H_{Min} = 2$ avec les autres codes. Les 7 cellules correspondantes sont en gris (dont une en gris clair pour le code lui-même ($h = 0$) et 6 autres en gris foncé pour $h = 1$). Les autres adresses correspondent à des codes qui ont une distance de Hamming $h \geq 2$ avec $A = \{2,2,2\}$ (en blanc, et en noir pour le code $\{0,0,0\}$).

3.3.3. Stratégie de recherche partagée

La distance de Hamming h apparaissant en tant qu'exposant dans (6), la méthode précédente est efficace pour des petites valeurs de H_{Min} . Pour des valeurs plus élevées, il est suggéré de décomposer h à l'aide de codes intermédiaires (Maurice *et al.*, 2012).

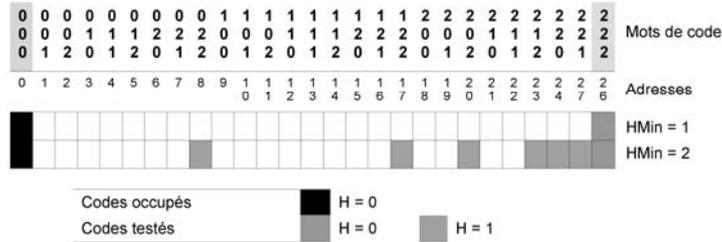


Figure 6. Illustration des H tests réalisés dans l'espace des adresses des codes

3.3.4. Résultats du codage optimisé

Le tableau 1 reporte les résultats obtenus pour $(u \times v) = (3 \times 3)$, $H_{Min} \in [1,5]$ et $C \in [2,8]$. Ces valeurs nous permettent de les comparer aux résultats de (Claes, 2008) (C) et de (Morano, 1998) (M). Avec les ressources disponibles (Pentium 4 à 3 Ghz, 2 Go RAM) les meilleurs résultats sont obtenus en quelques minutes avec $u_E = v_E = 5$. Dans tous les cas nous obtenons des matrices plus grandes que (M) (taille 45^2). Dans (X), nous arrivons en moyenne à générer des matrices sous-parfaites de taille équivalente avec 2 symboles de moins. Pour $H_{Min} \geq 2$, les résultats obtenus sont équivalents à ceux de (C) en termes de taille, mais pas en termes de temps de calculs où il faut plusieurs heures pour trouver des matrices avec une distribution non homogène des symboles.

Tableau 1. Résultats de générations de matrices sous-parfaites.

N.D. : non disponible, (M) : (Morano, 1998), (C) : (Claes, 2008)

k/H_{min}	1				2			3				4			5		
Note	Claes : N.D.												Morano : N.D.				
Algo	Nous <1s	Nous Max	C.	M.	Nous <1s	Nous Max	M.	Nous <1s	Nous Max	C.	M.	Nous <1s	Nous Max	M.	Nous <1s	Nous Max	C.
2	20 ²	22 ²	19x25	NA	10 ²	11 ²	NA	6 ²	7 ²	6x8	NA	5 ²	5 ²	NA	NA	NA	4x5
3	109 ²	111 ²	106x114	>= 45 ²	34 ²	37 ²	< 45 ²	15 ²	16 ²	14x18	< 45 ²	9 ²	9 ²	< 45 ²	6 ²	7 ²	6x8
4	397 ²	418 ²	316x421	>= 45 ²	103 ²	104 ²	< 45 ²	34 ²	37 ²	33x44	< 45 ²	16 ²	17 ²	< 45 ²	10 ²	10 ²	9x11
5	815 ²	1155 ²	429x572	>= 45 ²	242 ²	262 ²	>= 45 ²	67 ²	73 ²	68x90	< 45 ²	27 ²	27 ²	< 45 ²	12 ²	14 ²	12x16
6	855 ²	2580 ²	496x661	>= 45 ²	429 ²	548 ²	>= 45 ²	123 ²	133 ²	124x165	< 45 ²	41 ²	43 ²	< 45 ²	17 ²	20 ²	18x23
7	905 ²	5205 ²	676x901	>= 45 ²	445 ²	1023 ²	>= 45 ²	196 ²	228 ²	204x272	< 45 ²	44 ²	64 ²	< 45 ²	19 ²	27 ²	24x32
8	905 ²	9620 ²	696x925	>= 45 ²	365 ²	1645 ²	>= 45 ²	223 ²	344 ²	292x389	>= 45 ²	22 ²	92 ²	< 45 ²	14 ²	35 ²	32x42

Plus grande matrice, gain en taille : < 100% ou N.D. 100% - 500% > 500%

Plus grande matrice codée en moins de 1 sec. : N.D : Non Disponible

Pour $H_{Min} \leq 2$, notre approche est plus efficace. En effet, ici un comportement trop exhaustif mène à une explosion combinatoire alors que notre approche continue à obtenir des matrices de tailles qui augmentent quasi linéairement avec C.

De meilleurs résultats que ceux de la littérature ont été obtenus pour des constructions qui n'ont nécessité que quelques minutes d'exécution au plus. Il nous a été possible de générer des motifs de taille importante en une fraction de seconde (tableau 1, cellules "Nous ≤ 1 s") tout en autorisant un potentiel correctif dans le motif. Nous avons obtenu, en moins d'une seconde, des motifs de tailles plus importantes que celles reportées par (Adan, 2009). Finalement, notre algorithme génère des motifs matriciels avec moins de symboles en moyenne.

Il est à noter également que des motifs de tailles immenses ont été trouvés : 9620^2 pour $H_{Min} = 1$ et 1645^2 pour $H_{Min} = 2$ avec 8 symboles car pour de faibles valeurs de H_{Min} la mise en œuvre des H tests est la plus efficace. L'augmentation sans cesse des résolutions autorisera l'exploitation de tels motifs dans quelques années. Mais dès aujourd'hui, la génération rapide (en quelques secondes seulement) de grands motifs permet de tester différents jeux de paramètres et de déterminer celui qui est le mieux approprié. Enfin, nous souhaitons mentionner qu'il est possible de concevoir des motifs en temps réel (vidéo - 25 i/s). Ceci permettra à l'avenir d'adapter le motif à la scène observée (adapter les indices visuels ou augmenter localement la résolution...), sur une séquence d'images.

4. Conception de motifs basée sur la géométrie épipolaire

Dans cette section, nous allons d'abord introduire la géométrie épipolaire puis les sous-matrices sous-parfaites pour finalement présenter la conception de motifs non matriciels à l'aide d'un alignement des indices visuels sur des droites épipolaires. Pour cela, la caméra et le projecteur sont supposés obéir au modèle de la projection perspective.

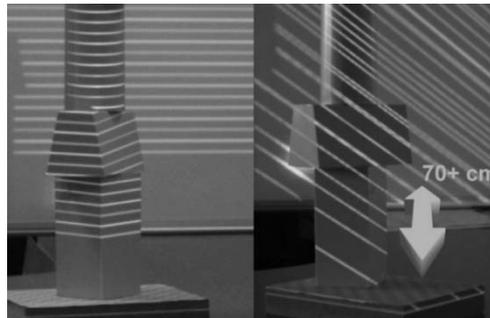


Figure 7. Projection de lignes sur le PIC. (Gauche) Les lignes sont toutes horizontales dans le PIP, leur projection sur le PIC dépend de la scène. (Droite) Les lignes passent par l'épipoles dans le PIP, leur projection sur le PIC est indépendante de la scène et se fait uniquement le long des lignes épipolaires associées. Cette invariance reste valable sur une grande plage de profondeur : plus de 70 cm ici

4.1. Géométrie épipolaire

Il est établi depuis longtemps que, connaissant les coordonnées d'un point dans une des images d'un système stéréoscopique (le projecteur ou la caméra ici), la géométrie épipolaire (Longuet-Higgins, 1981 ; Hartley, Zisserman, 2000) permet de retrouver la ligne épipolaire correspondante dans l'autre plan. La contrainte épipolaire est couramment utilisée dans les systèmes stéréo pour réduire l'espace de recherche durant l'appariement des indices visuels. Si elle permet de contraindre l'information de positionnement des indices qui doivent s'apparier, ce n'est pas le cas de l'information d'orientation. En effet, dans le cas général, si la scène à reconstruire est inconnue, il n'est pas possible de savoir comment une ligne projetée sur la scène, se comporte dans le plan image de la caméra (PIC). Cependant, il existe une singularité : si dans le plan image du projecteur (PIP) des lignes passent par l'épipole, alors leur projection dans le PIC correspondra à la ligne épipolaire correspondante (intersection entre le plan épipolaire et le PIC). Ainsi, quelle que soit la scène considérée, cette projection restera invariante aux variations dans la scène (figure 7). Pour que cette invariance reste valide sur une large profondeur de champ et puisse offrir un domaine de validité important, l'estimation de la matrice Fondamentale – qui contient toute la géométrie épipolaire – doit être suffisamment précise.

La méthode proposée s'appuie sur l'homographie dégénérée qui relie les droites épipolaires correspondantes entre les deux plans images (figure 8). Ainsi, pour un ensemble de droites épipolaires P_p qui intersectent l'épipole e_p dans le plan image du projecteur, les droites épipolaires correspondantes dans le plan image de la caméra, P_c , intersectent l'épipole e_c selon la relation où apparaît la matrice fondamentale F :

$$P_p \equiv F [e_c]_{\times} P_c. \quad (7)$$

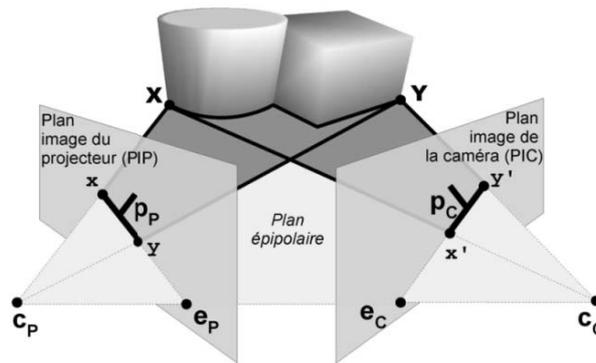


Figure 8. Géométrie d'un système stéréo utilisant le modèle sténopé. c_p , e_p , P_p , x et y : respectivement, le centre optique, l'épipole, un indice orienté selon la ligne épipolaire passant par x, y et e_p sur le PIP. c_c , e_c , P_c , x' et y' : les équivalents pour le PIC. X et Y sont les triangulations de x, x' et y, y' sur la scène 3-D

4.2. Conception du motif final

La recherche des voisinages dans le PIC constitue la principale difficulté avec le codage spatial. Si ceci demeure facile en présence de scènes planes, pour des scènes quelconques cette recherche est loin d'être triviale. C'est pourquoi le motif final est conçu à partir de la connaissance de la géométrie épipolaire qui caractérise la relation projective, que nous avons auparavant supposée constante, entre le projecteur et la caméra. La finalité est donc de transférer *dans* le motif lui-même les relations géométriques liant le PIC et le PIP afin de déterminer immédiatement les indices visuels contribuant au même voisinage. Généralement, les indices visuels sont régulièrement répartis dans le PIP, sur une grille par exemple, puis ils sont recherchés dans le PIC en fonction de la proximité des lignes épipolaires. Pour tenir compte des imprécisions géométriques résiduelles, la recherche est menée au sein d'une zone d'une certaine largeur dans laquelle plusieurs autres indices, *a priori* non connus, mais ayant une ligne épipolaire proche, peuvent apparaître. Cependant il est possible de savoir exactement quels indices visuels géométriques vont être ou non en concurrence lors de l'appariement en les disposant directement sur les lignes épipolaires suffisamment distinctes dans le PIP. Cela a un double intérêt. D'une part, il est possible de savoir exactement, lors du codage, entre quels indices la contrainte H_{Min} doit porter. D'autre part, cela simplifie la segmentation, les indices visuels étant regroupés par ligne épipolaire dans le PIP.

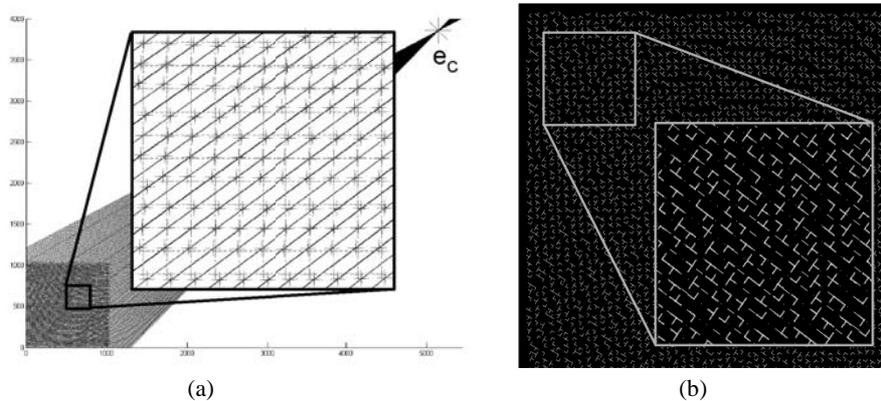


Figure 9. Conception d'un motif (50×50) dépendant de la géométrie épipolaire.
 (a) Placement des indices visuels depuis une grille sur les lignes épipolaires les plus proches (noir) dans le PIP. e_c : l'épipole dans le PIP.
 (b) Un motif final dans le PIP (Résolution : 50^2 , $C = 8$, $H_{Min} = 6$)

Pour disposer les indices visuels par ligne épipolaire tout en gardant une distribution homogène de ceux-ci dans le PIP et le PIC, nous proposons l'approche suivante, décomposée en quatre étapes, et illustrée sur la figure 9.

1. Choisir dans le PIC des lignes épipolaires régulièrement réparties, le long desquelles on souhaite que les indices visuels se projettent. Leur nombre et espacement dépendront de la résolution du motif et de la marge que l'on souhaite garder pour limiter les risques d'ambiguïté lors de la segmentation.

2. Identifier les lignes épipolaires correspondantes dans le PIP grâce à la matrice fondamentale, celles-ci doivent couvrir la surface du motif.

3. A partir d'une grille régulière ayant la résolution du motif final souhaitée, projeter orthogonalement chaque intersection de la grille (croix) sur la ligne épipolaire la plus proche, identifiée à l'étape 2. Les positions obtenues (croix) correspondent aux centres des indices dans le PIP.

4. A l'issue de cette étape, les sous-ensembles d'indices visuels entre lesquels la distance H_{Min} doit être respectée sont ceux dont les centres se trouvent sur la même ligne épipolaire.

4.3. Codage du motif

Les sous-ensembles étant établis, il est possible de coder le motif avec l'algorithme décrit précédemment (section 3.2), sauf que la distance H_{Min} n'est plus vérifiée entre tous les voisinages (3×3) du motif mais seulement entre les voisinages ayant un indice central se trouvant sur la même ligne épipolaire. En d'autres termes, la recherche effectuée pour les H Tests n'est ici que *locale*, correspondant à une catégorie de matrices, appelées sous-matrices sous-parfaites (*SubPerfect SubMaps* ou SPSM). Une SPSM est donc générée. Ceci permet d'alléger considérablement les contraintes et de fournir des motifs ayant une distance de Hamming minimum H_{Min} plus élevée, à résolution et nombre de symboles identiques, que pour les motifs basés sur les matrices sous-parfaites (PSM). Il en résulte une capacité correctrice plus importante lors du décodage.

4.4. Projection des motifs

Nous illustrons ici les avantages du codage réalisé sur le motif à projeter à l'aide d'un ensemble de reconstructions. Nous avons choisi de représenter les symboles avec des indices visuels cunéiformes pour projeter un motif monochromatique (figure 10). Après avoir corrigé les distorsions géométriques et estimé la géométrie épipolaire, les indices visuels cunéiformes ont été orientés selon des lignes épipolaires afin de faciliter la segmentation et d'éviter de recourir à une invariance rotationnelle des indices, comme c'est le cas dans (Claes, 2008) et (Adan, 2009).

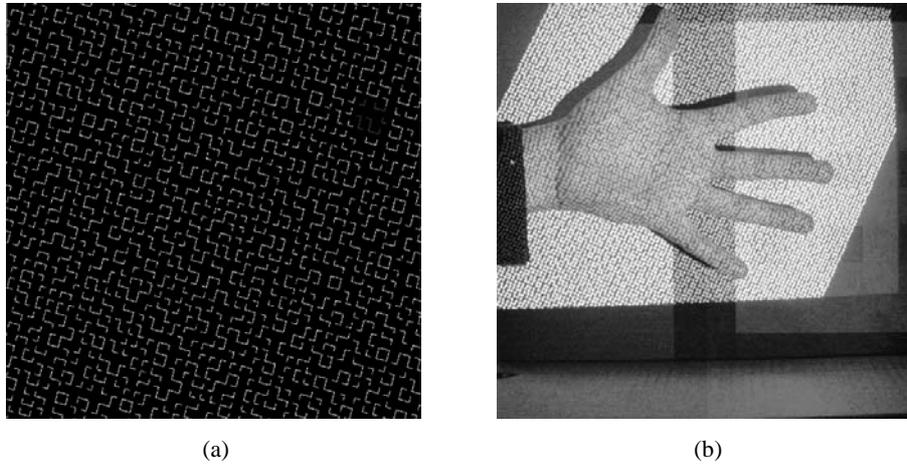


Figure 10. (a) Motif (100×150) aligné sur des droites épipolaires, avec $H_{Min} = 4$ et $C = 4$ symboles (associés à des indices visuels cunéiformes).
 (b) Même motif projeté sur une main

5. Reconstruction de surfaces à partir d'une image

Pour toutes les reconstructions présentées ci-après, un motif de (100×150) avec 4 symboles et $H_{Min} = 4$ a été employé (comme sur la figure 10). Les reconstructions ont été obtenues, grâce à une parallélisation spécifique des traitements sur GPU, en moins de 40 ms.

Sur la figure 11, la sensibilité du codage vis-à-vis de diverses perturbations géométriques ou spectrales est faible ; les surfaces sont reconstruites (figure 11b, d) sans inclure des imprécisions autres que celles dues à la résolution employée.

La figure 12 montre l'intérêt d'effectuer la reconstruction à partir d'une seule image pour un objet en mouvement, comme une main qui se referme. La position des doigts, malgré les occultations qu'ils produisent est bien restituée. En effet, grâce aux capacités d'autocorrections intégrées, un indice visuel totalement occulté peut néanmoins être identifié si son voisinage (et les voisinages adjacents) contient suffisamment d'informations pour retrouver son code. D'ailleurs, sur la figure 13, on peut observer les différences dans la qualité de reconstruction de la main selon la résolution du motif projeté ((100×150) et (50×50)).

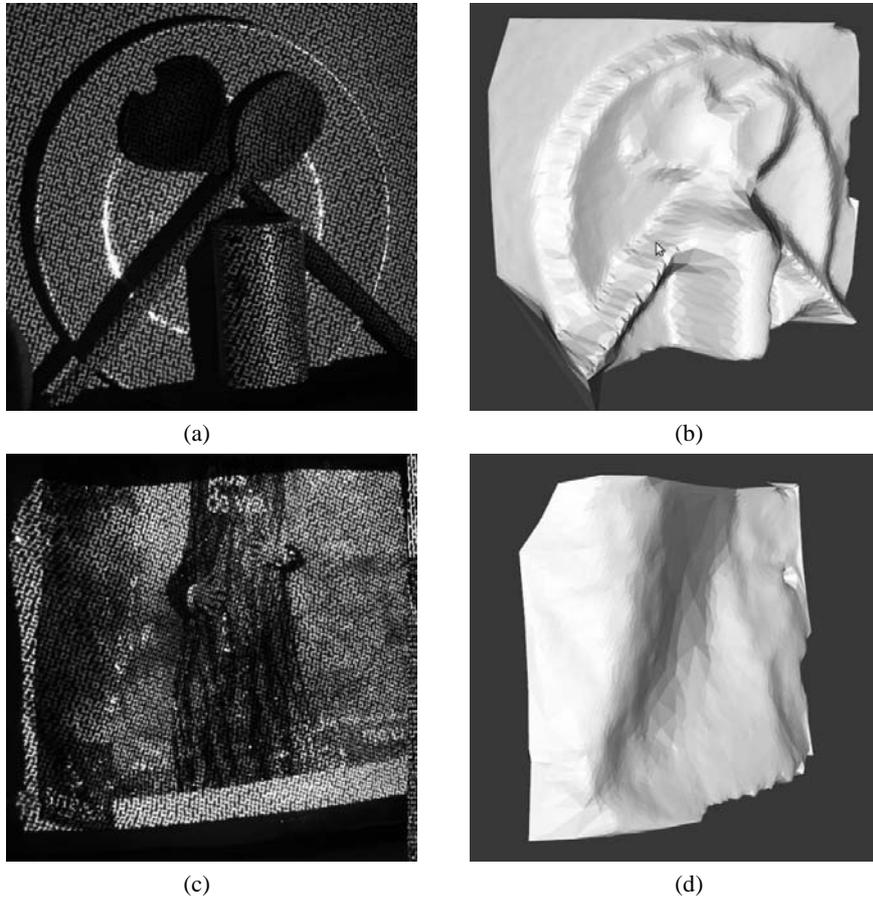


Figure 11. Sensibilité du décodage vis-à-vis de diverses perturbations.
(a,b) : occultations, (c,d) : perturbations spectrales

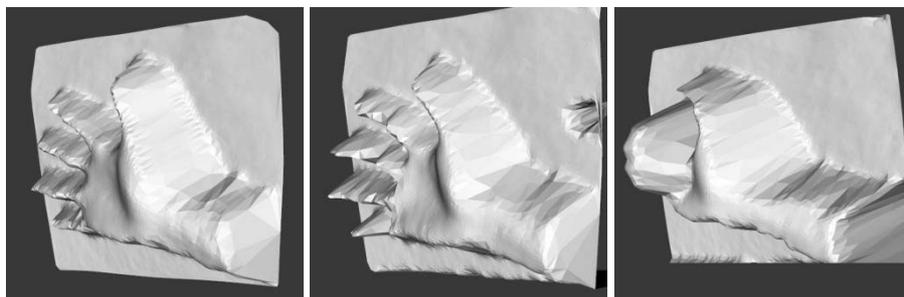


Figure 12. Reconstruction en temps réel d'une main se refermant

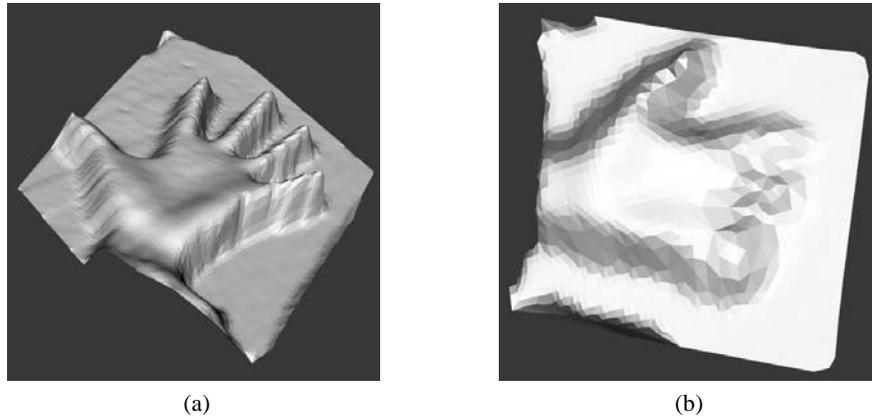


Figure 13. Précision de la reconstruction en fonction de la résolution du motif projeté : (a) avec un motif (100×150), (b) avec un motif (50×50)

6. Conclusion

Nous avons présenté dans cet article un nouvel algorithme de conception de motifs pour la vision par projection de lumière structurée. Pour pallier le manque de données dans la capture des motifs projetés, des capacités de correction doivent y être intégrées (distance de Hamming minimum supérieure à 1). Mise en œuvre à l'aide d'algorithmes de type « brute-force », la construction informatique de motifs souffre d'une complexité algorithmique élevée, qui limite la réalisation pratique des motifs de grande taille. Pour réduire ce caractère aussi bien lors du codage que du décodage, nous avons présenté une étude statistique qui montre que la distance de Hamming moyenne est une caractéristique globale qui permet d'établir le choix des paramètres du futur motif (nombre de symboles, longueur des codes, distance de Hamming minimum) en fonction du potentiel correctif que l'on souhaite intégrer dans le motif. Par la suite, nous avons proposé des optimisations en vue de réduire le nombre de tests sur la distance de Hamming (*H Tests*). Ces tests conditionnent les performances des algorithmes de construction de motifs et ont une influence de plus en plus importante quand la taille du motif augmente. Parmi ces optimisations, nous avons présenté une méthode qui met en œuvre les tests dans l'espace 1-D des adresses des codes plutôt que dans l'espace 2-D de la matrice du motif. Pour des valeurs élevées de la distance de Hamming, nous avons proposé une répartition qui tient compte des tests déjà réalisés, pour capitaliser les efforts de calculs. Les résultats sur le codage confirment la réduction de la complexité algorithmique.

Une méthode exploitant la géométrie épipolaire a été présentée par la suite, et s'appuyant sur les algorithmes de type brute-force, permet de construire des motifs non matriciels, où les indices visuels sont positionnés le long des droites épipolaires. Ceci simplifie notablement le codage et le décodage qui sont alors locaux et permettent de guider les algorithmes de segmentation et de classification des indices

visuels associés aux symboles du code. Une mise en œuvre sur scène complexe avec de nombreuses difficultés typiques a montré la robustesse de tels motifs et l'intérêt de la méthode. Finalement, des reconstructions de surfaces obtenues en temps réel à partir d'une seule image démontrent le potentiel applicatif et ouvrent une perspective intéressante, celle des motifs qui pourront s'adapter, en temps réel, à des variations géométriques et photométriques de la scène observée.

Bibliographie

- Adan A., Vázquez A., Cerrada C., Salamanca S. (2009). Moving surface extraction based on unordered hexagonal perfect submaps projection: Applications to 3D feature tracking. *Image and Vision Computing*, vol. 27, n° 8, p 1083-1096.
- Albitar C., Graebbling P., Doignon C. (2007). Robust structured light coding for 3D reconstruction. *IEEE Int'l Conference on Computer Vision*, Rio de Janeiro, Brésil.
- Battle J., Mouaddib E.M., Salvi J. (1998). *Recent progress in coded structured light as a technique to solve the correspondence problem: a survey*. *Pattern Recognition*, vol. 31, n° 7, p. 963-982.
- Chan S.Y., Li Y.F., Zhang J. (2007). Real-time Structured Light Vision with the Principle of Unique Color Codes. *IEEE Int'l Conf. on Robotics and Automation*, p. 429-434.
- Claes K. (2008). *Structured Light Adapted to Control a Robot Arm*. Thèse de Doctorat, Katholik Univ. Louvain, Belgique.
- De Bruijn N. G. (1946). A combinatorial problem. *Nederlandske Akad. van Wetenschappen*, p. 758-764.
- Etzion T. (1988). Constructions for perfect maps and pseudorandom arrays, *IEEE Transactions on Information Theory*, vol. 34, n° 5/1, p. 1308-1316.
- Griffin P.M., Narasimhan L.S., Yee S.R. (1992). Generation of uniquely encoded light patterns for range data acquisition, *Pattern Recognition*, vol. 25, n° 6, p. 609-616.
- Güthring J. (2001). Dense 3-D surface acquisition by structured light using off-the-shelf components. *Videometrics and Optical Methods for 3D Shape Measurement*, vol. 4309, p. 220-231.
- Hartley R., Zisserman A. (2000). *Multiple view geometry in computer vision*. Cambridge Univ. Press.
- Hulbert G., Garth I. (1993). On the de Bruijn torus problem. *Journal of Combinatorial Theory Ser. A*, vol. 64, p. 50-62.
- Hung D. (1993). 3D scene modeling by sinusoid encoded illumination. *Image and Vision Computing*, vol. 11, p. 251-256.
- Ishii I., Yamamoto K., Doi K. and Tsuji T. (2007). High-speed 3D image acquisition using coded structured light projection. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, p. 925-930, St Louis, USA.
- Kawasaki H. Furukawa R., Sagawa R., Yagi Y. (2008). Dynamic scene shape reconstruction using a single structured light pattern. *IEEE Int'l Conf. on Computer Vision and Pattern Recognition*, Anchorage, Alaska, USA.

- Koninckx T.P., Van Gool L. (2006). Real-Time Range Acquisition by Adaptive Structured Light. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, n° 3, p. 432-445.
- Longuet-Higgins C. (1981). A computer algorithm for reconstructing a scene from two projections, *Nature*, vol. 293, p. 61-62.
- Maurice X., C. Doignon (2012). Méthodologie de conception dirigée par la distance de Hamming : application à la reconstruction de surfaces à partir d'une seule image, *Actes de la conférence RFIA*, Lyon.
- Maurice X., Graebling P., Doignon C. (2011). A pattern framework driven by the Hamming distance for structured light-based reconstruction with a single image. *IEEE Int'l Conference on Computer Vision and Pattern Recognition*, Colorado Springs, USA.
- Maurice X., Graebling P., Doignon C. (2010). Conception et appariement de motifs géométriques pour une projection de lumière structurée. *Actes de la conférence RFIA*, Caen, France.
- Mitchell C.J. (1995). Aperiodic and semi-periodic perfect maps. *IEEE Transactions on Information Theory*, vol. 41, n° 1, p. 88-95.
- Morano A., Ozturk C., Conn R., Dubin S., Zietz S., Nissanov J. (1998). Structured Light Using Pseudo-random Codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, n° 3, p. 322-327.
- Morita, H., Yajima, K., Sakata, S. (1988). Reconstruction of surfaces of 3-D objects by M-Array pattern projection method. *IEEE Int'l. Conference on Computer Vision*, vol. 88, p. 468-473.
- Pagès J., Salvi J. Collewet C., Forest J. (2005). Optimised De Bruijn patterns for one-shot acquisition. *Image and Vision Computing*, vol. 23, p. 707-720.
- Patterson K.G. (1994). Perfect Maps. *IEEE Transactions on Information Theory*, vol. 40, n° 3, p. 743-753.
- Peterson W.W., Weldon E.J. Jr. (1972). *Error-Correcting Codes*, 2nd edition, Cambridge MIT Press.
- Pribanic T., Dapo H., Salvi J. (2009). Efficient and Low-Cost 3D Structured light system based on a modified number-theoretic approach. *EURASIP Journal on Advances in Signal Processing*, vol. 2010.
- Sagawa R., Ota Y., Yagi Y., Furukawa R., Asada N. (2009). Dense 3-D reconstruction method using a single pattern for fast moving object. *IEEE Int'l Conf. Computer Vision*, Kyoto, Japon.
- Salvi J., Pagès J., Batlle J. (2004). Pattern codification strategies in structured light systems. *Pattern Recognition*, vol. 37, p. 827-849.
- Salvi J., Batlle J., Mouaddib E.M. (1998). A robust-coded pattern projection for dynamic 3d scene measurement. *Pattern Recognition Letters*, vol. 19, p. 1055-1065.
- Singleton R.C. (1964). Maximum distance q-nary codes. *IEEE Transactions on Information Theory*, vol. 10, p. 116-118.

- Weise T., Leibe B., Van Gool L. (2007). Fast 3-D scanning with automatic motion compensation. *IEEE Int'l Conf. on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA.
- Young M., Beeson E., Davis J., Rusinkiewicz S., Ramamoorthi R. (2007). View-point coded structured light. *IEEE Int'l Conf. on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA.
- Zhang L., Curless B., Seitz S. (2002). Rapid shape acquisition using structured light and multi-pass dynamic programming, *Int'l Symp. on 3D Data Processing, Visualization and Transmission*, p. 24-36.

Xavier Maurice est docteur de l'Université de Strasbourg, spécialité signal, image, automatique et robotique depuis décembre 2011. Sa thèse portait sur la reconstruction 3-D de scènes intra-abdominales par l'utilisation de lumière structurée codée. En 2008, il a reçu le diplôme d'ingénieur de l'INSA de Lyon, spécialité informatique. En 2006, il a co-réalisé Virtual-Drums : www.virtual-drums.com. Il s'est spécialisé en traitement d'images et vision par ordinateur et a contribué au projet européen « 3D Anatomical Human », pendant une année d'échange à l'EPFL, Suisse. Il s'intéresse principalement à la réalité virtuelle/augmentée, la vision par ordinateur, la modélisation 3-D. Aujourd'hui, ses recherches portent sur le suivi 3-D de visages humains.

Christophe Doignon est docteur de l'Université Louis Pasteur de Strasbourg (1994). Après avoir effectué un séjour post-doctoral à l'université de Padoue (Italie) en 1995, il intègre le laboratoire LSIT de l'Université Louis Pasteur en 1996 en tant que maître de conférences. En 2007, il obtient l'habilitation à diriger les recherches, puis un poste de professeur à Télécom Physique Strasbourg en 2012, pour rejoindre le laboratoire ICube (UMR 7357) de l'Université de Strasbourg. Ses activités de recherche scientifique concernent la vision par ordinateur, les asservissements visuels et la commande de robots, le traitement du signal et des images en temps réel, pour des applications relevant principalement de l'assistance aux gestes chirurgicaux par l'imagerie et la robotique médicales.