HAL open science

Génération automatique de motifs symétriques pseudo-aléatoires pour les applications de la lumière structurée codée

Xavier Maurice, Pierre Graebling, Christophe Doignon

To cite this version:

Xavier Maurice, Pierre Graebling, Christophe Doignon. Génération automatique de motifs symétriques pseudo-aléatoires pour les applications de la lumière structurée codée. ORASIS'09 - Congrès des jeunes chercheurs en vision par ordinateur, 2009, Trégastel, France, France. inria-00404612

HAL Id: inria-00404612

https://inria.hal.science/inria-00404612v1

Submitted on 16 Jul 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Génération automatique de motifs symétriques pseudo-aléatoires pour les applications de la lumière structurée codée

An efficient algorithm for building symmetrical pseudo-random patterns with applications to coded structured lighting

Xavier MAURICE Pierre GRAEBLING Christophe DOIGNON

Université de Strasbourg
Laboratoire des Sciences de l'Image, de l'Informatique et de la Télédétection (LSIIT)
Equipe Automatique, Vision et Robotique
Pôle API - Bd Sébastien Brant, 67412 Illkirch
{xavier.maurice,graebling,doignon}@lsiit.u-strasbg.fr

Résumé

Dans cet article, nous proposons un algorithme de génération automatique de motifs symétriques pour des applications en vision active utilisant la lumière structurée codée. De tels motifs sont notamment utilisés pour la reconstruction 3D et la modélisation de surfaces. L'algorithme présenté ici utilise une approche par voisinage, à l'aide des matrices parfaites, pour identifier de façon unique chaque voisinage et permettre une mise en correspondance rapide des primitives géométriques associées aux symboles. Nous considérons la distance de Hamming comme un des paramètres de l'algorithme et comme critère de robustesse permettant de comparer plusieurs approches entre elles.
Notre contribution principale se situe dans l'approche suivie pour la construction de la matrice du motif global, en particulier par la capitalisation des sous-ensembles de solutions trouvées pour les motifs élémentaires et par la réduction de la complexité algorithmique. Il devient alors possible, tout en utilisant les mêmes ressources (nombre de symboles, taille du voisinage et taille du motif global) de trouver des solutions avec une distance de Hamming supérieure. Ceci permet d'assurer le décodage et la correction si nécessaire même dans le cas où un grand nombre de primitives sont manquantes, l'algorithme proposé offre alors une meilleure robustesse au bruit et aux occultations.

Mots Clef

Lumière structurée codée, vision active, matrices pseudo-aléatoires, analyse de scènes 3D.

Abstract

In this paper, we propose a new algorithm for the automatic generation of symetrical patterns designed for active vision applications using coded structured lighting. Such patterns are used in fields like 3D reconstruction and surface modelisation. A neighbourhood approach is used, thanks to perfect maps, to uniquely identify the code of the sub-matrix surrounding interest points - the pattern elements - and to allow a quick mapping of the geometrical primitives associated to each alphabet's component. The Hamming distance is used as an input parameter as well as a robustness criterion for the comparison of several approaches.
Our main contribution concerns the building of the entire pattern, and is related to the capitalization capabilities on subsets of found solutions for pattern elements and the algorithmic complexity reduction of the process. Then, the proposed algorithm is able to find solutions with a higher Hamming distance, while using the same resources (alphabet, neighbourhood and pattern sizes). Consequently, the decoding and correction can be ensured even if an important number of geometrical primitives are missing. The proposed solution offers a better robustness to noise and occlusions.

Keywords

Coded structured lighting, active vision, pseudorandom matrices, 3D scene analysis.

1 Introduction

1.1 Contexte

L'utilisation de la vision pour le guidage de robots dans les environnements complexes se heurte souvent aux difficultés liées à l'interprétation des images. C'est le cas, par exemple, en laparoscopie robotisée où le positionnement des instruments de chirurgie vis-à-vis des organes est contrôlé à l'aide d'une caméra endoscopique. Les récentes avancées en chirurgie mini-invasive et chirurgie par voies naturelles ont permis l'insertion complète d'instruments dans le corps d'un patient et d'effectuer des opérations chirurgicales réduisant les traumastismes post-opératoires.

Cependant l'opérateur ne peut plus avoir une vue directe sur la zone d'intervention et la plupart du temps un système de vision articielle est mis en place pour offrir au chirurgien la meilleure représentation visuelle possible.
L'assistance visuelle au geste chirurgical est ainsi devenue un domaine de recherche en plein essor, accompagnant souvent l'assistance robotique. Il s'agit entre autre d'élaborer des techniques performantes pour interpréter convenablement les apparences visuelles des parois intra-abdominales dans le but, soit d'étendre la représentation visuelle (réalité augmentée) par la reconstruction 3D et la modélisation des surfaces internes, soit de mettre en place un guidage robotisé (asservissement visuel) [7].

Parmi toutes les techniques existantes, la projection de lumière structurée sur la scène à analyser et l'interprétation d'une ou plusieurs images de cette projection est un bon moyen de restituer lŠinformation 3D relative à cette scène. En effet, par rapport à un dispositif plus classique, comme un dispositif stéréoscopique par exemple, il permet dŠintroduire une information connue à la scène et ainsi de faciliter la reconstruction de zones très homogènes et faiblement texturées, là où les dispositifs classiques ont plus de difficultés. Cependant, de nombreuses sources de perturbations (occultations, fortes courbures de surfaces, spécularités, zones hors-champ, etc..) ne permettent pas toujours d'imager lŠensemble de la projection et ainsi assurer une mise en correspondance entre l'ensemble des motifs projetés et ceux détectés dans l'image. Afin de résoudre ce problème de mise en correspondance avec seulement un nombre restreint de primitives identifiées, il est nécessaire d'appliquer un codage aux primitives du motif.

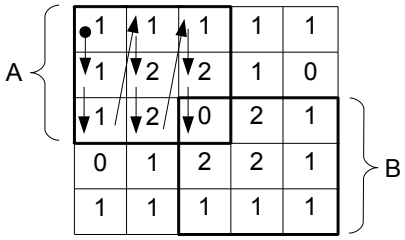
Parmi les stratégies de codage résumées par J. Salvi dans [12], celles qui relèvent du multiplexage temporel ne permettent pas d'envisager une reconstruction en temps réel de scènes en mouvement, celles caractérisées par un codage direct sont trop sensibles aux changements d'illumination de la scène et finalement une approche par voisinage spatial avait été retenue.
Dans une publication précédente [1], nous avions alors proposé un nouveau motif de lumière structurée qui utilisait les matrices parfaites - chaque sous-matrice d'une taille donnée n'apparaît qu'une seule fois - dans le but d'identifier de façon unique (et donc résoudre le problème de correspondance) et rapide les motifs élémentaires correspondant à des voisinages 3×3 autour d'une primitive géométrique. La discrimination des symboles par la géométrie des indices visuels avait été préférée à une discrimination par la couleur ou l'intensité, d'une part, pour réduire la sensibilité face aux variations de luminosité et de couleurs de la scène et d'autre part pour satisfaire les contraintes imposées par la miniaturisation du dispositif. L'approche employant la lumière monochromatique permet d'envisager une utilisation en chirurgie mini-invasive grâce à une source laser de couleur verte (couleur majoritairement absente dans les scènes intra-abdominales), une fibre optique et l'emploi d'optiques diffractives pour fabriquer le motif global.
La présence d'une primitive directionnelle, le trait, dans le motif présente deux avantages. Il fournit une information d'orientation amenant à la génération de motifs élémentaires distincts par rotation, ce qui permet de ne pas réduire le nombre de solutions possibles pour les motifs élémentaires à déterminer par la suite, contrairement à ce qui est proposé dans [3]. D'autre part, la connaissance de l'orientation locale autour de cette primitive guide la recherche de la position des primitives voisines, facilitant leur détection.
Par ailleurs, une symétrie centrale a été respectée pour faire un meilleur usage de l'énergie lumineuse incidente sur les optiques diffractives, permettant d'exploiter jusqu'à 80 % de l'énergie du rayonnement laser contre 40 % autrement.
C'est ainsi qu'une matrice de dimensions 27×29 utilisant un alphabet de 3 symboles : un trait, un disque et un cercle a été proposée dans [1].
Pour évaluer la robustesse de notre motif global, nous avions utilisé comme critère la distance de Hamming (nombre de positions en lesquels les symboles de deux mots de code diffèrent) entre les mots de code associés aux motifs élémentaires. Nous obtenions une valeur moyenne de cette statistique de 6.0084, assurant un pouvoir de discrimination important entre les mots de codes utilisés. Cependant, elle ne garantit pas une valeur minimale de la distance de Hamming. Rien n'indique que pour tout mot de code obtenu, il n'en existe pas un autre qui ait avec lui une distance de Hamming de 1. On ne pouvait alors pas garantir un décodage sans ambiguïté dans le cas de primitives manquantes.

Terminologie et notations Le motif global est représenté par une matrice G composée de symboles s ; c'est la mire. Un symbole est représenté ici par une primitive géométrique. Un motif élémentaire entoure chaque symbole par d'autres symboles pour former un voisinage. Souvent carré ou rectangulaire, on parle alors de fenêtre de voisinage ou de sous-matrice, mais il peut être structuré autrement. La taille de l'alphabet est représenté par A. Le nombre de composants du motif élémentaire correspond à la longueur du mot de code et est noté W. Le code S est défini par la séquence de parcours des symboles au sein du motif élémentaire. Sur la figure 1, le code du motif élémentaire A entourant une primitive située à la position (i, j) correspond à la séquence de parcours

$$\begin{aligned} S_{ij} \quad &= \{s_{i-1,j-1}, s_{i,j-1}, s_{i+1,j-1}, s_{i-1,j}, s_{i,j}, \\ &\qquad s_{i+1,j}, s_{i-1,j+1}, s_{i,j+1}, s_{i+1,j+1}\} \end{aligned} \tag{1}$$

Par la suite, w représente la largeur d'une fenêtre carrée, ses dimensions sont plus généralement notées $p \times q$ et W, la longueur du mot de code associé ($W = p\ q$). h représente la distance de Hamming (minimale) du motif global, c'est à dire la plus petite distance de Hamming entre tous les couples de mots de code de taille de W possibles. Dans unematrice dite parfaite, a donc nécessairement

$h \geq 1$. L et M représentent les dimensions de la matrice représentant le motif global. Ceci est illustré par la figure 1.



Code de A : 111122120
Code de B : 021221111

FIGURE 1 – Une matrice parfaite, symétrique, avec $A = 3$, $p = q = w = 3$, $h = 5$ et $L = M = 5$. A et B sont deux motifs élémentaires de distance de Hamming 6, B est le symétrique central de A. Le code de B correspond à la séquence inverse des symboles du code de son symétrique A.

1.2 Contributions et structure de l'article

La présente contribution s'intéresse au processus de génération *automatique* d'un motif global avec une distance de Hamming minimale imposée, devant aboutir à des possibilités de décodage et de correction supérieures lorsque l'ensemble des primitives d'un voisinage n'a pas pu être décodé dans son intégralité. Il s'agit également de fournir un nouvel algorithme suffisamment rapide et efficace pour générer des matrices de grandes dimensions supportant les capacités mémoires des ordinateurs actuels, dans un laps de temps raisonnable.
Dans la suite de l'introduction, les objectifs souhaités sont précisés. Dans la section suivante, un état de l'art des solutions existantes est dressé. L'architecture de l'algorithme proposé est présentée dans la section 3 et les résultats sont comparés à l'état de l'art actuel, en particulier avec ceux obtenus par Morano [8] et ceux obtenus par Claes [3] dans la section 4. Une conclusion accompagnée des perspectives envisagées est présentée dans l'ultime section.

1.3 Objectifs

S'agissant de la génération de matrices pseudo-aléatoires parfaites dans le but de fournir un motif de lumière structurée pour la reconstruction 3D de surfaces ou le guidage par leurs apparences dans les images, quatre paramètres, considérés indépendamment les uns des autres, sont identifiés pour comparer deux motifs :

1. **la distance de Hamming** : pour augmenter le pouvoir discriminant du motif global, celle-ci doit être maximale. En effet, si des primitives n'ont pu être identifiées pour une raison ou une autre, une valeur élevée de la distance de Hamming offre plusieurs possibilités de décodage sans ambiguïté.
2. **la taille de l'alphabet** : il doit être de taille minimale, car utiliser le moins de primitives possibles permet de les définir le plus dissemblablement possible
3. **la taille du motif élémentaire** : elle sera choisie de préférence avec une valeur faible, sinon le nombre de possibilités à tester devient rapidement prohibitif (les autres paramètres étant fixés). Avec peu de primitives pour caractériser un motif élémentaire, plus de motifs élémentaires projetés seront identifiés et finalement plus de points seront reconstruits dans les zones à fortes variations de surfaces. Par la suite, la modélisation des surfaces pourra faire appel localement à un voisinage 3D (ou un maillage) formé d'un plus grand nombre de points de contrôle.
4. **la taille de la matrice** : celle-ci doit être maximale si l'objectif est lié à l'obtention d'une plus grande résolution et par conséquent à une meilleure précision dans la reconstruction et la localisation 3D. Il y a tout de même une limitation physique car des images de primitives trop petites peuvent poser des problèmes durant la segmentation et la classification.

Nous admettrons que, pour trois de ces paramètres imposés, l'algorithme permettant de construire des matrices maximisant le 4ème paramètre sera considéré comme meilleur que les autres.
Le second objectif de ce travail est d'aboutir à la fois à une réduction de la complexité algorithmique du processus de génération de matrices et de proposer une implémentation efficace pour réduire les temps de calculs, notamment pour l'obtention de matrices à grandes dimensions. Bien que moins déterminant, il conditionne la possibilité d'offrir un outil pragmatique.

Relation entre correction et distance de Hamming : Le motif global à projeter constitue un signal connu précisément et caractérisé par une distance de Hamming de valeur égale à h. Si $h-1$ symboles ne sont pas détectés sur un motif élémentaire, il restera toujours un symbole supplémentaire permettant de discriminer ce motif élémentaire de tous les autres et de retrouver l'unique code complet correspondant dans la liste des codes du signal d'entrée. Ainsi, connaissant le motif à projeter, pour être capable d'identifier de façon unique et de corriger le code associé à tout motif élémentaire dont n symboles n'ont pas pu être détectés après projection, il est nécessaire de concevoir un motif global caractérisé par une distance de Hamming supérieure ou égale à $n+1$.

2 Travaux antérieurs

Le problème de la génération des matrices pseudo-aléatoires parfaites a commencé à être étudié il y a de nombreuses années. Nous pouvons classer les approches en deux catégories : les approches analytiques, directes et les approches de type *brute-force*. Les approches analytiques,

directes, reposent sur d'importants fondements théoriques, souvent sur les séquences de De Bruijn [4] (équivalent monodimensionnel des matrices parfaites), mais ne permettent d'obtenir que des matrices de tailles limitées [5] ou disproportionnées [9] et dans tous les cas, ne permettent pas d'assurer une distance de Hamming h supérieure à 1 ([2, 6]). Par contre, des conditions d'existence ont pu être émises par Patterson *et al.* dans [10] et Reed *et al.* dans [11], mais ne concernent que le cas $h = 1$.

Pour les approches de type *brute-force*, utilisées dans [3, 9, 8], des éléments sont ajoutés au fur et à mesure pour remplir le motif global. A chaque nouvel ajout, une vérification est effectuée pour s'assurer que la distance de Hamming entre les nouveaux codes générés par l'apparition de ce nouvel élément et l'ensemble des codes déjà connus, soit supérieure ou égale à la distance de Hamming minimale désirée (1 dans le cas des matrices parfaites). Si ce n'est pas le cas pour tous les symboles possibles pour cet élément, différentes stratégies - qui seront discutées dans les sections suivantes - sont possibles pour guider au mieux la recherche d'une nouvelle solution.

Ces dernières approches permettent de dépasser les limites que rencontrent les méthodes analytiques et fournissent, de façon générale, de meilleurs algorithmes, dans le cadre des objectifs discutés plus tôt. Par contre, à part la limite supérieure du nombre total de mots de code possibles, A^W, la théorie ne permet pas encore de savoir si pour p, q, A, L, M et h donnés, une matrice parfaite de taille $L \times M$ existe ou non. Seule une recherche exhaustive permet de le savoir. Cependant, dès que ces paramètres deviennent importants, l'explosion combinatoire engendrée ne permet pas de tout tester. De par nos besoins applicatifs, c'est vers ce type d'approches que nous nous sommes tournés.

Dans Morano *et al.* [8], un algorithme entièrement pseudo-aléatoire est proposé. Un premier motif 3×3 (figure 2-a) de la matrice est généré pseudo-aléatoirement. Puis les 3 premières lignes (figure 2-b) sont remplies de gauche à droite avec un nouveau motif élémentaire incluant 3 nouveaux éléments (sur la même colonne). Les 3 premières colonnes sont remplies de façon analogue (figure 2-c). A partir du premier élément libre (4ème ligne et 4ème colonne), chaque nouvel élément introduit génère un nouveau motif élémentaire 3×3 (figure. 2-d). A chaque fois qu'un nouvel élément est généré pseudo aléatoirement, chaque nouveau motif élémentaire créé est comparé à tous ceux déjà trouvés pour vérifier la valeur de la distance de Hamming. Si celle-ci est supérieure ou égale à h, l'algorithme passe à l'élément suivant, sinon tous les symboles possibles sont essayés pour cet élément. Si aucun ne convient, l'algorithme reprend depuis le début.

Dans Claes *et al.* [3], une matrice est générée sans que sa taille ne soit déterminée à priori. Partant d'un seul motif élémentaire, la matrice croît au fur et à mesure en ajoutant des lignes et des colonnes, puis en remplissant le reste de la matrice, de telle sorte que le rapport hauteur/largeur, choisi arbitrairement, reste constant. Cela permet d'obtenir un résultat à chaque essai, sans à priori sur la taille de la matrice. Par opposition à l'approche pseudo-aléatoire employée par Morano, celle-ci est plus proche d'une recherche exhaustive. Les auteurs interprètent la suite d'éléments trouvés comme un très grand nombre à incrémenter construit à partir de polynômes en base A. Ainsi à chaque position correspond une pondération (figure 3). Cela permet, d'une part, de ne tester qu'une seule fois chaque configuration matricielle, ce qui n'est pas assuré dans l'approche pseudo-aléatoire. D'autre part, dans le cas où aucun symbole ne convient pour l'élément recherché (valeur de Hamming pré-définie non obtenue) comme par exemple dans la figure 3, le symbol rouge, qui complète le motif élémentaire S, , cela permet de capitaliser sur tout ce qui a déjà été trouvé précédemment en incrémentant l'élément correspondant au chiffre de poids supérieur dans le grand nombre généré et en recommençant la recherche sur cet élément, après avoir remis l'élément courant à 0 (contrairement à Morano qui recommence complètement la recherche). Comme dans l'application envisagée dans [3], aucune contrainte des mouvements de la caméra par raport au projecteur n'est imposée, cette démarche aboutit à des motifs élémentaires qui vérifient l'invariance par rotation pour des rotations de $\pi/2$, π et $3\pi/4$, car le motif global proposé n'inclut pas de primitive directionnelle. L'avantage par rapport à l'algorithme de Morano est que la matrice obtenue n'est pas entièrement regénérée dès qu'un motif élémentaire ne peut

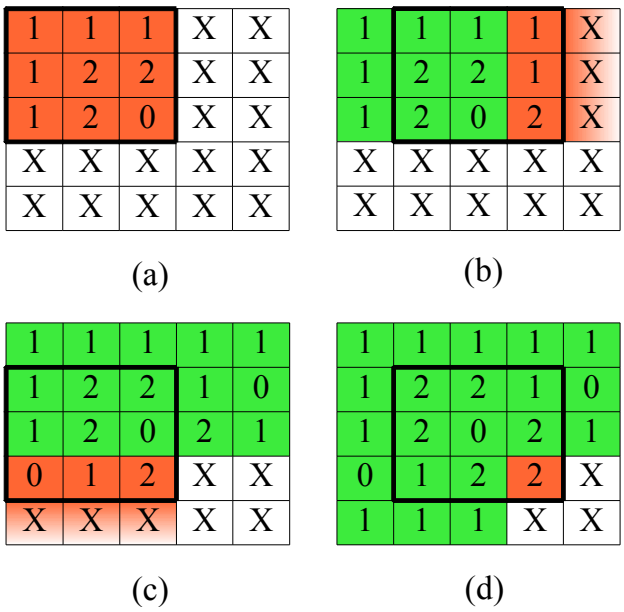


FIGURE 2 – Remplissage du motif global selon Morano. (En vert) : éléments déjà trouvés. Un nouveau motif élémentaire est créé (en gras) lorsque, (a) : le motif élémentaire en haut à gauche est généré aléatoirement, (b) : les trois premières lignes sont remplies, par chaque nouveau triplet d'éléments (orange) sur la même colonne, (c) les trois premières colonnes sont remplies, par chaque nouveau triplet d'éléments (orange) sur la même ligne, (d) par chaque nouvel élément (orange).

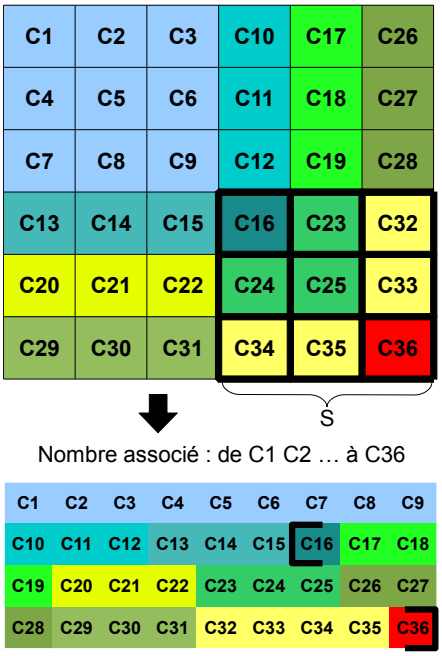


FIGURE 3 – Remplissage selon Claes. Le motif global croît au fur et à mesure. A partir d'un motif élémentaire (bleu clair), des colonnes et lignes sont ajoutées (triplets de même couleur) et le reste de la matrice est complété. Lors d'un blocage sur un symbole (C36, en rouge), le nombre associé de C1 (poids fort) à la position de ce symbole est incrémenté. Les crochets représentent le nombre qu'il faut avoir atteint par des incrémentations successives, à partir de C36 (poids faible), pour modifier tous les éléments du motif élémentaire (en gras).

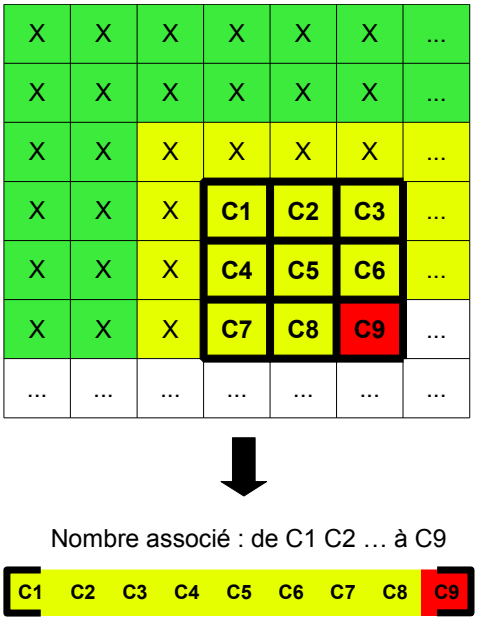


FIGURE 4 – Dans l'approche que nous proposons, uniquement les éléments constituant le dernier motif élémentaire créé (en gras) de C1 à C9, et le symbole en situation de blocage (en rouge), contribuent à définir le nombre associé (entre crochets) qui sera incrémenté.

trouver de symbole qui convienne dans l'alphabet. Tout ce qui a été trouvé précédemment est réutilisé. Par contre cette approche montre ses limites dès que les paramètres A, L, M et w deviennent importants. En effet, le nombre en base A dont il est question est tellement grand que, quand un certain nombre de motifs élémentaires a déjà été trouvé, seulement les quelques monômes de poids faibles peuvent être modifiés par l'incrémentation, car le nombre d'essais nécessaires pour atteindre les éléments de poids forts de ce nombre est tellement grand qu'en pratique seulement 1, voire 2 lignes de symboles sont explorées. Ceci limite les possibilités de correction et se ressent dans les résultats obtenus avec $h = 5$, $w = 3$. L'utilisation de $A = 7$ ou $A = 8$ symboles ne permet pas d'améliorer les résultats déjà obtenus avec $A = 6$ symboles en un temps raisonnable.

3 Algorithme proposé

L'approche que nous avons décidé d'adopter reprend la séquence de remplissage et emprunte l'aspect pseudo-aléatoire de Morano en l'associant à une variation judicieuse de l'approche par capitalisation et incrémentation de Claes. Contrairement à [3], nous n'utilisons pas de contrainte d'invariance par rotation car la position relative entre le projecteur et la camera reste la même dans notre cas. Cela évite de réduire l'espace des solutions possibles. Nous présentons tout d'abord l'algorithme de remplissage du motif global et la façon dont est gérée la contrainte de symétrie centrale que nous avons évoquée précédemment. Nous présentons ensuite notre gestion des situations bloquées (contrainte h non respectée). Finalement, dans le cas où $h = 1$, nous présentons une méthode permettant, pour chaque nouveau motif, d'obtenir un test de présence à accès direct, ramenant la complexité algorithmique de $\mathrm{O}(LM)$ à $\mathrm{O}(1)$ pour les algorithmes précédents.

3.1 Algorithme de remplissage avec prise en compte de la symétrie centrale

Initialement, le motif global de taille arbitrairement choisie est complètement rempli de façon pseudo-aléatoire avec une distribution uniforme, de façon à fournir une entropie importante. En effet, une situation de départ avec une variance importante, favorise d'emblée une grande distance de Hamming entre les motifs élémentaires déjà présents. Dans le cas contraire, il y a une probabilité plus élevée de trouver des motifs élémentaires semblables.

Par ailleurs, du fait de l'emploi envisagé des optiques diffractives pour la fabrication, un élément central de forme circulaire doit être présent pour représenter la composante continue du faisceau laser, la symétrie centrale impose alors que les deux dimensions du motif global, L et M correspondent à des nombres impairs.

Pour faciliter la compréhension, les règles de construction décrites ici sont données pour $w = 3$, mais elles peuvent être étendues pour tout motif élémentaire isomorphe par symétrie centrale et possédant un élément central (forme en losange par exemple).

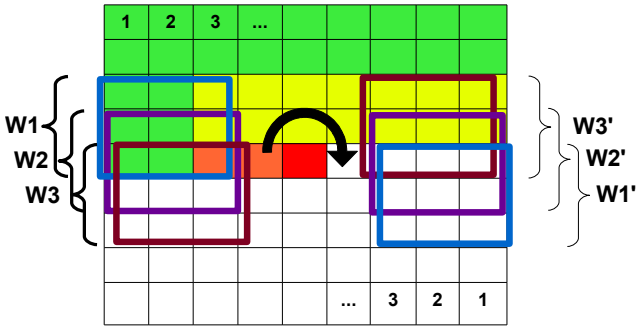


FIGURE 5 – Génération d'une matrice parfaite symétrique pour $w = 3$. Dès qu'un nouveau symbole est trouvé, le même symbole est utilisé pour son symétrique (éléments : 1, 2, 3). Chaque élément des deux premières lignes / colonnes (zone verte), ne génère aucun nouveau motif élémentaire. Chaque élément L de la ligne du milieu (zone orange) génère 3 nouveaux motifs élémentaires (W1, W2, W3) et son symétrique en génère 3 autres (W1', W2', W3') symétriques des trois premiers. Le symbole central, (indiqué en rouge) génère 9 nouveaux motifs. L'algorithme remplissant la matrice de gauche à droite et de haut en bas, (pour se terminer sur l'élement central), chaque symbole restant (zone jaune) complète un motif élémentaire dont il représente l'élément en bas à droite. Son symétrique est également créé.

Comme le motif global doit être à symétrie centrale, à chaque fois qu' un nouvel élément, de position (i, j) dans la matrice, est introduit, il est également associé à l'élément occupant la place symétrique centrale en position $(L - i, M - j)$. Cette correspondance est illustrée par les éléments numérotés 1, 2, 3 sur la figure 5). Ainsi, il n'est nécessaire de rechercher que des symboles pour la moitié des éléments (en couleur sur la figure 5), auxquels s'ajoutera un élément central. Comme la matrice est construite symétriquement ligne par ligne, pour $w = 3$, les éléments des deux premières lignes et des deux premières colonnes (en vert sur la figure 5), ne contribuent pas à créer de nouveaux motifs élémentaires, et donc aucun test n'est nécessaire pour eux.

Pour la ligne centrale (en orange sur la figure 5, chaque nouvel élément crée 3 nouveaux motifs élémentaires et leurs symétriques (sur figure 5, l'élément L crée W1, W2, W3 et son symétrique crée W1', W2', W3') . Il y a donc six vérifications à effectuer avec tous les codes des motifs élémentaires déjà présents dans le motif global, chaque vérification étant faite en O(LM) tests. Mais il faut aussi vérifier que les codes de ces six nouveaux motifs élémentaires ont entre eux une distance de Hamming $\geq h$, ce qui nécessite 15 nouveaux tests . L'élément central (en rouge sur la figure 5) contribue à créer 9 nouveaux motifs élémentaires (4 plus leurs 4 symétriques plus le motif élémentaire central, seul motif élémentaire isomorphe par symétrie centrale). Comme précédemment, les codes associés aux nouveaux motifs élémentaires doivent être vérifiés avec les codes des motifs déjà trouvés, et entre eux. Pour tous les autres éléments (en jaune sur la figure 5, un nouveau motif élémentaire et son symétrique sont créés. En plus des tests sur le respect de la distance de Hamming il faut également effectuer la vérification entre les codes de ces deux nouveaux motifs.

Si pour chaque nouvel élément, l'ensemble des tests permet d'assurer la contrainte sur h, cet élément ainsi que son symétrique sont conservés et l'algorithme passe à l'élément suivant. Sinon les autres symboles sont testés. Si une solution émerge avec un de ces autres symboles, l'algorithme passe à l'élément suivant, sinon, le mécanisme de capitalisation, présenté ci-après, est utilisé pour continuer la croissance du motif global.

3.2 Recherche par capitalisation

Lorsque pour un nouvel élément, aucun symbole ne convient, cela signifie que parmi les éléments de la matrice déjà trouvés, se trouve un motif élémentaire dont le code a une distance de Hamming avec le code du nouveau motif élémentaire créé inférieure à h. Pour résoudre ce conflit, soit il faut chercher des modifications dans le nouveau motif élémentaire, soit dans l'ancien. L'ancien motif élémentaire se trouvant forcément plus en amont que le nouveau, si des modifications y sont apportées, il faudra regénérer tous les éléments du motif global depuis l'élément modifié dans l'ancien motif élémentaire. Pour capitaliser au mieux sur tout ce qui a déjà été trouvé et ne pas avoir à regénérer une partie importante du motif global, il est donc plus intéressant de chercher à modifier le nouveau motif. Claes, dans son approche, incrémente le nombre représenté successivement par tous les symboles trouvés depuis le début (figure 3), mais ce nombre est tellement grand que seulement les quelques derniers chiffres de ce grand nombre sont réellement incrémentés dans un laps de temps raisonnable. Pour pouvoir accéder à l'élément C16 du nouveau motif afin de l'incrémenter, il faudra d'abord avoir testé, par incrémentations successives tous les symboles possibles du nombre composé de 20 chiffres de C17 à C36, ce qui peut être est excessivement long, notamment quand la taille de l'alphabet est importante. De plus, plus la matrice grandit plus la taille du nombre à incrémenter pour tester toutes les configurations du nouveau motif élémentaire est importante, ce qui fait exploser le nombre de tests nécessaires.

Dans notre implémentation, au lieu de considérer ce grand nombre, nous ne considérons que le nombre, composé de W chiffres, correspondant au dernier motif élémentaire, de l'indice C1 à C9 (voir la figure 4). Comme les valeurs de départ des chiffres composant ce nombre sont connues (résultats de la première génération aléatoire de toute la matrice), nous pouvons faire prendre à ce nombre toutes les valeurs possibles en faisant varier par incrémentation chacun de ses éléments de sa valeur initiale : i à $(i + A - 1)$ modulo A. Quand tous les symboles possibles ont été épuisés pour un élément, le chiffre correspondant à cet élément est remis à sa valeur initiale et l'élément correspondant au chiffre de poids supérieur dans le nombre étudié est incré-

A / h	1		2		3		4		5	
Algo	Claes	Nous	Claes	Nous	Claes	Nous	Claes	Nous	Claes	Nous
2	11x14	17x17	NA	9x9	4x5	5x7	NA	5x5	3x4	NA
3	28x37	81x81	NA	29x29	9x12	13x13	NA	9x9	4x5	7x5
4	70x93	291x291	NA	65x65	11x14	25x25	NA	13x13	5x7	9x7
5	177x236	NA	NA	111x111	23x30	47x47	NA	21x21	8x10	13x11
6	205x270	NA	NA	NA	44x58	81x81	NA	31x31	10x13	15x15
7	297x396	NA	NA	NA	57x75	NA	NA	45x45	10x13	17x17
8	316x421	NA	NA	NA	75x100	NA	NA	55x55	10x13	21x21

FIGURE 6 – Tailles maximales obtenues par la génération de matrices pseudo-aléatoires parfaites, pour différentes valeurs des paramètres A (nombre de symboles) et de h (distance de Hamming), avec $w = 3$. NA : non disponible.

menté à son tour.

Cette approche nous permet d'être certains d'avoir étudié toutes les combinaisons possibles pour ce motif élémentaire en un temps limité et indépendamment du nombre d'éléments déjà trouvés dans la matrice.

Au pire, si les éléments C1, C2 ou C3 (figure 4) de ce motif élémentaire sont modifiés, seulement les deux dernières lignes du motif global seront à regénérer. Si aucune solution n'a pu été trouvée à l'issue cette étape de correction du dernier motif élémentaire, la recherche est relancée depuis le début avec une nouvelle génération pseudo-aléatoire.

L'approche que nous suggérons propose un équilibre entre une exploration purement aléatoire, comme le proposede Morano[8], et une recherche exhaustive, comme le propose Claes[3]. Cette façon d'appréhender le problème permet dŠexplorer des zones très éloignées dans lŠespace des solutions tout en analysant localement, là où cela peut être interessant, des petites zones de façon exhaustive pour trouver une solution qui respecte h. Nous verrons dans les résultats que cette approche permet de trouver de nouvelles solutions plus intéressantes que ce qui a été trouvé par Morano et Claes.

3.3 Réduction de la complexité algorithmique pour $h = 1$

Les algorithmes de type *fill-and-test-as-you-go-along* nécessitent un grand nombre de tests pour s'assurer de l'unicité des motifs élémentaires nouvellement incorporés. Chaque nouveau motif élémentaire n'est valide que si, après comparaison avec l'ensemble des autres motifs élémentaires déjà trouvés, aucun doublon n'a été repéré. Si n motifs élémentaires ont déjà été trouvés, à cette vérification correspond une complexité en O(n). Nous présentons ici une méthode permettant d'effectuer la même vérification en accès direct, en O(1), réduisant considérablement la complexité algorithmique et nous ayant permis de trouver des matrices parfaites de tailles inégalées avec le même alphabet et des motifs élémentaires de même taille. Le principe de la méthode proposée est très simple : il consiste à stocker, pour chaque mot de code possible, dans un tableau monodimensionnel, un drapeau binaire qui indique si le code est disponible (drapeau à 0 ou à 1). Grâce à un stockage ordonné, l'adresse à laquelle est stocké directement le drapeau d'un code dans le tableau, correspond à la version décimale de ce code.

Le nombre de combinaisons de mots possibles de longueur W avec A symboles est égal à A^W. Il est possible d'énumérer de façon exhaustive toutes ces possibilités en commençant par le mot de code représenté par tous les symboles à 0 puis en l'incrémentant en base A jusqu'à obtenir le mot de code représenté que par des "$A - 1$". Pour passer d'un mot de code au suivant, il suffit donc d'ajouter 1 au nombre que représente ce mot de code. Nous observons qu'il en va de même pour désigner l'adresse, dans le tableau de tous les codes possibles. Ainsi, une bijection existe entre l'ensemble des mots de code possibles et leur adresse de stockage. Cette dernière peut alors être calculée simplement, si les mots de codes sont stockés dans l'ordre du nombre qu'ils représentent, en convertissant ce nombre A-aire en décimal.

Calcul de l'adresse du drapeau d'un mot de code. Etant donné un code C de longueur W utilisant A symboles. C peut être assimilé à un nombre formé de W chiffres en base A. Si on note par C_n le n^{ieme} chiffre du code C en partant du poids fort (C_1), alors l'adresse du code dans le tableau de l'ensemble des mots possibles est donnée par l'expression suivante

$$\text{Adresse}(C) = \sum_{n=1}^{n=W} C_n \, A^{W-n} \, . \qquad (2)$$

Les ressources nécessaires pour les cas étudiés ont été, au pire pour le cas $A = 8$, $9^8 = 134$ drapeaux. Comme un bit est suffisant pour coder chaque drapeau 16 Mo de mémoire vive ont suffit.

4 Résultats

Tous les résultats sont fournis pour une taille de fenêtre $w = 3$ et n'ont nécessité que quelques minutes de temps de calculs sur ordinateur PC.

Dans la figure 6, nous avons rassemblé les résultats obtenus avec la méthode de Claes et la notre. Pour cela, les deux algorithmes respectifs ont essayé de générer la plus grande

A / h	1			2			3			4		
Algo	Morano	Claes	Nous	Morano	Claes	Nous	Morano	Claes	Nous	Morano	Claes	Nous
3	v	v	v	x	NA	x	x	x	x	x	NA	x
4	v	v	v	x	NA	v	x	x	x	x	NA	x
5	v	v	v	v	NA	v	x	x	v	x	NA	x
6	v	v	v	v	NA	v	x	x	v	x	NA	x
7	v	v	v	v	NA	v	x	v	v	x	NA	v
8	v	v	v	v	NA	v	v	v	v	x	NA	v

FIGURE 7 – Génération d'une matrice pseudo-aléatoire parfaite de taille 45×45, pour différentes valeurs des paramètres A et h. (v : une matrice a pu être trouvée, x : aucune matrice n'a pu être trouvée, NA : non disponible).

matrice possible, pour différentes valeurs des paramètres A et h. En analysant le tableau des résultats, nous constatons que, dans tous les cas, notre algorithme génère des matrices plus grandes. Trois facteurs principaux nous ont permis de parvenir à cela :

1. la configuration projecteur-caméra utilisée nous a permis de ne pas poser l'invariance par rotation comme contrainte, permettant de conserver un espace de solutions plus grand.
2. une utilisation plus judicieuse de la capitalisation des codes des motifs élémentaires déjà trouvés pour sortir des situations de blocage.
3. dans le cas où $h = 1$, l'utilisation de l'accès direct, qui nous permet d'explorer un nombre de solutions largement plus élevé dans un même laps de temps.

Tout comme dans les méthodes de Morano [8], et Claes [3], nous avons étudié les valeurs des paramètres A et h qui permettent de générer une matrice de taille 45×45 (figure 7), afin de savoir quel algorithme permet de générer une matrice d'une telle taille avec le moins de symboles possibles. Les résultats obtenus montrent que dans tous les cas l'algorithme que nous proposons nécessite moins (pour $h > 1$) ou autant de symboles (pour $h = 1$) que les autres algorithmes.

Nous notons en particulier que dans le cas d'une distance de Hamming de 3, notre algorithme réussit à générer la matrice, présentée sur la figure 8 avec seulement 5 symboles, alors que l'algorithme de Morano dans [8] en nécessite 8 et celui de Claes en nécessite 7 dans [3].

De plus, nous observons que dans les mêmes conditions que [1] (et pour 3 symboles considérés), une matrice de même taille (29×29) mais avec une distance de Hamming supérieure ($h = 2$ au lieu de 1) est générée (figure 9), ce qui pourra permettre, en utilisant le même nombre de symboles, de décoder des motifs avec une primitive manquante.

Enfin, pour montrer l'apport de la contrainte sur h, différentes possibilités de correction associées à la mire sont étudiées (figure 9), avec différentes valeurs de h dans le cas de primitives localisées mais non identifiées. Nous pouvons noter que dans cet exemple le taux de primitives décodées en plus par rapport à $h = 1$, est de : $h = 2$: 12%,$h = 3$: 28%,$h = 4$: 102%,$h = 5$: 110%, ce qui est très significatif.

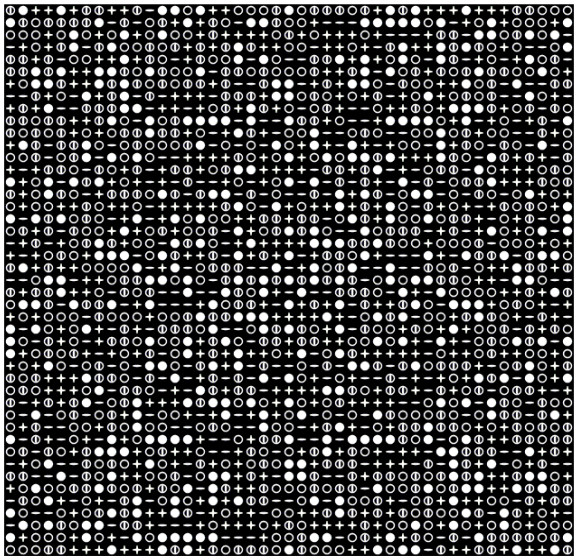


FIGURE 8 – Exemple de génération d'une matrice pseudo-aléatoire parfaite G de taille $L = M = 45$ avec seulement 5 symboles différents ($w = 3$ et $h = 3$).

5 Conclusion et perspectives

Nous avons présenté dans cet article un nouvel algorithme de génération automatique de matrices pseudo-aléatoires parfaites symétriques dont la la distance de Hamming est imposée. Cet algorithme, grâce à une capitalisation judicieuse des solutions intermédiaires trouvées et à une réduction drastique de la complexité algorithmique pour une distance de Hamming de 1, a permis de générer, en quelques minutes, des résultats meilleurs que ceux présents dans l'état de l'art. Grâce aux solutions trouvées, une plus grande robustesse lors du décodage des motifs en environnements complexes est attendue.

Cependant des contraintes différentes supplémentaires, l'invariance par rotation dans [3] et la symétrie centrale dans notre cas, ont été imposées. Les comparaisons sont donc faites à titre indicatif, et il serait intéressant de comparer les algorithmes dans des conditions strictement identiques. Dans tous les cas, ces deux algorithmes donnent des résultats meilleurs que dans [8], qui respecte pourtant

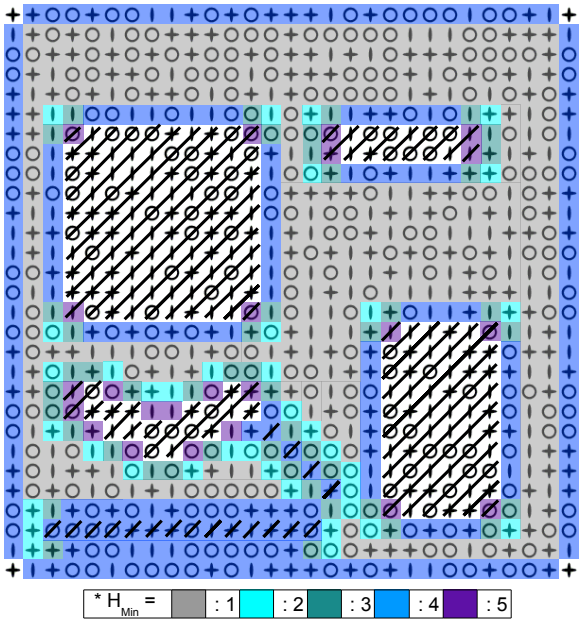


FIGURE 9 – Exemple de motif symétrique produit (taille 29^2,$A = 3$, $w = 3$ et $h = 2$). Les éléments hachurés représentent des exemples de primitives localisées mais non identifiées. Les couleurs indiquent les primitives au centre des nouveaux voisinages 3×3 pouvant être décodés par correction si ce motif avait $h = 1$ à 5 (*).

moins de contraintes.

Références

[1] C. Albitar, P. Graebling, and C. Doignon. Nouveau motif pour une reconstruction 3d en vision active codée. In *Actes de la conférence francophone RFIA*, Amiens, Janvier 2008.

[2] S.Y. Chen, Y.F. Li, and J. Zhang. Realtime structured light vision with the principle of unique color codes. In *IEEE Int'l Conf. on Robotics and Automation*, pages 429–434, Rome, Avril 2007.

[3] K. Claes and H. Bruyninckx. Robot positioning using structured light patterns suitable for self calibration and 3d tracking. In *Int'l Conf. on Advanced Robotics*, Jeju, Korea, Août 2007.

[4] N. G. de Bruijn. A combinatorial problem. In *Nederlanske Akademie van Wetenschappen*, number 49, pages 758–764, 1946.

[5] T. Etzion. Constructions for perfect maps and pseudorandom arrays. *IEEE Transactions on Information Theory*, 34(5/1) :1308–1316, 1988.

[6] A. Kazantsev and E. M. Petriu. Robust pseudorandom coded colored structured light technique for 3d object model recovery. In *IEEE Int'l Workshop on Robotic and Sensors Environments*, pages 150–155, Ottawa, Octobre 2008.

[7] A. Krupa, J. Gangloff, C. Doignon, M. de Mathelin, G. Morel, L. Soler, J. Leroy, and J. Marescaux. Autonomous retrieval and 3d positioning of surgical instruments in robotized laparoscopic surgery. *IEEE Transactions on Robotics and Automation*, 19(5) :842–853, Octobre 2003.

[8] A. Morano, C. Ozturk, R. Conn, S. Dubin, S. Zietz, and J. Nissanov. Structured light using pseudorandom codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3) :322–327, 1998.

[9] C. Ozturk, J. Nissanov, and S. Dubin. Generation of perfect map codes for an active stereo imaging system. In *IEEE Annual Northeast Bioengineering Conference*, pages 76–77, 1996.

[10] K. G. Patterson. Perfect maps. *IEEE Transactions on Information Theory*, 40(3) :743–753, 1994.

[11] I. S. Reed and R. M. Stewart. Note on the existence of perfect maps. *IRE Transactions on Information Theory*, 8(1) :10–12, Janvier 1962.

[12] J. Salvi, J. Pagès, and J. Batlle. Pattern codifications strategies in structured light systems. *Pattern Recognition*, 37 :827–849, 2004.