



HAL
open science

Méthodologie de conception de motifs dirigée par la distance de Hamming : application à la reconstruction de surfaces à partir d'une seule image

Xavier Maurice, Christophe Doignon

► To cite this version:

Xavier Maurice, Christophe Doignon. Méthodologie de conception de motifs dirigée par la distance de Hamming : application à la reconstruction de surfaces à partir d'une seule image. RFIA 2012 (Reconnaissance des Formes et Intelligence Artificielle), Jan 2012, Lyon, France. pp.978-2-9539515-2-3. hal-00656515

HAL Id: hal-00656515

<https://hal.science/hal-00656515>

Submitted on 17 Jan 2012

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.

Méthodologie de conception de motifs dirigée par la distance de Hamming : application à la reconstruction de surfaces à partir d'une seule image

Xavier MAURICE

Christophe DOIGNON

Université de Strasbourg
LSIIT (UMR CNRS-UdS 7005)

Equipe Automatique, Vision et Robotique

IRCAD - 1, Place de l'Hôpital, 67091 Strasbourg cedex

Courriels : xavier.maurice@etu.unistra.fr , 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 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. A cette fin, nous avons développé ici une méthode de conception de motifs codés basée sur la distance de Hamming, critère dont la valeur minimale peut être spécifiée dès la conception. 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. Nous montrons que la distance de Hamming moyenne est un critère statistique pertinent pour prédire les propriétés globales du futur motif, au-delà de l'unicité des codes qui le constituent, et nous fournissons un moyen de la calculer. Afin de réduire la complexité algorithmique, l'unicité des codes est examinée dans l'espace mono-dimensionnel des adresses des codes et non dans l'espace 2-D du motif matriciel, comme c'est le cas dans la littérature. Les résultats montrent une réduction significative de la complexité 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 3-D de scènes diverses, réalisées à partir d'une seule image des motifs projetés sont reportées, illustrant le potentiel applicatif de notre approche.

Mots Clef

Mise en correspondance, lumière structurée codée, reconstruction 3-D, distance de Hamming.

Abstract

Structured light based patterns provide a means to capture the state of an object shape. However it may be inefficient when the object is freely moving, when its surface contains high curvature parts or in out of depth of field situations.

For image-based robotic guidance in unstructured and dynamic environment, 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. To this end, we have developed a novel technique for the generation of coded patterns directly driven by the Hamming distance. The counterpart is the big amount of codes the coding/decoding algorithms have to face with a high desired Hamming distance. We show that the mean Hamming distance is a useful criterion for driving the patterns generation process and we give a way to predict its value. Furthermore, to ensure local uniqueness of codewords 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 codeword 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. Finally, real-time reconstructions from single images are reported and results are compared to the best known which are outperformed in many cases.

Keywords

Correspondence problem, coded structured light, 3-D reconstruction, Hamming distance.

1 Introduction

La projection de lumière structurée est basée sur l'apport de signaux prédéfinis 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. Par le passé, la vision par projection de lumière structurée de lignes, de primitives colorées ou de plusieurs niveaux d'intensité, a été largement utilisée pour acquérir et numériser des surfaces d'objets statiques ou en mouvement contrôlé ([13], [8], [17], [22], [4], [10]).

Dans le cas d'objets se déplaçant librement, Weise *et al.* [20] à l'aide de la stéréovision et du décalage de phase, et Young *et al.* [21] avec la projection d'un réseau de droites,

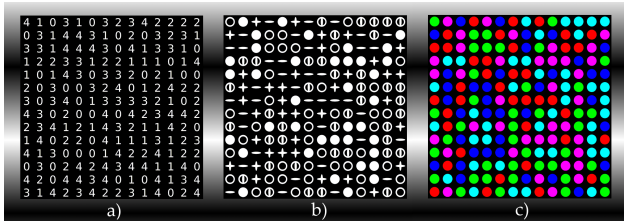


FIGURE 1 – Un motif 5-aire, 14×14 basé sur les matrices parfaites, d'une distance de Hamming de 5 entre tous les codes (voisinages 3×3) et représenté à l'aide a) de nombres, b) de formes, c) de couleurs. Au moins 2 erreurs par voisinage peuvent être corrigées.

ont proposé des techniques de reconstruction de surfaces à partir d'une seule acquisition d'image. Dans [9], Kawasaki *et al.* réalise une reconstruction dense des surfaces grâce à l'utilisation des informations d'adjacence locale au sein d'une grille de motifs. Récemment, en 2009, Sagawa *et al.* a 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 [16]. Toutes ces techniques nécessitent d'associer un indice visuel à un symbole particulier 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 navigation et l'interaction avec des organes dans la cavité abdominale par endoscopie, nous avons développé une technique de reconstruction en temps réel de surfaces qui fait appel à la projection de lumière structurée. A cette fin, nous avons retenu les approches appartenant à la famille des voisinages spatiaux. En effet, celles relevant du multiplexage temporel ne sont pas adaptées à la reconstruction de scènes dynamiques et les méthodes employant le codage direct s'avèrent beaucoup trop sensibles aux changements de luminosité ou des perturbations spéculaires pour être utilisées de manière fiable dans ce contexte. Dans la catégorie des techniques par voisinage spatial, un code identifie le voisinage (une sous-matrice le plus souvent) par l'agencement des 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 ([3],[18]). 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 (d'intensité) de gris (Fig. 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 ([1],[11]). La précision d'une reconstruction 3-D obtenue par projection de lumière structurée est liée 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.

1.1 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. D'autre part, 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. Une matrice parfaite est l'équivalent 2-D des séquences de De Bruijn [6] qui ont été très largement employées en lumière structurée ([15], [17]). Etant donné une taille de voisinage ($W = u \times v$), la difficulté est d'obtenir une matrice parfaite aux dimensions $m \times n$ souhaitées. Afin d'atteindre des valeurs élevées, Pagès [14] a utilisé un codage hybride basé sur les séquences de De Bruijn et composé d'un codage dans l'espace RVB et d'un codage du niveau de gris. A notre connaissance, les premières constructions directes de motifs basés sur les matrices parfaites ont été proposées par Etzion *et al.* [7] dans le cas binaire ($C = 2$). Les algorithmes ne permettaient 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. 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. Morano [12] et plus récemment Albitar *et al.* [2] à l'aide d'indices visuels géométriques et Claes [5] avec des spots colorés ont proposé des constructions de motifs basées sur les algorithmes "brute-force". 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.* [1] a employé les PSM pour 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, on constate en pratique que la construction de motifs peut nécessiter plusieurs heures de calculs pour des motifs de grandes tailles. Les dimensions des motifs étant donc limitées, il en résulte une précision de reconstruction 3-D limitée également.

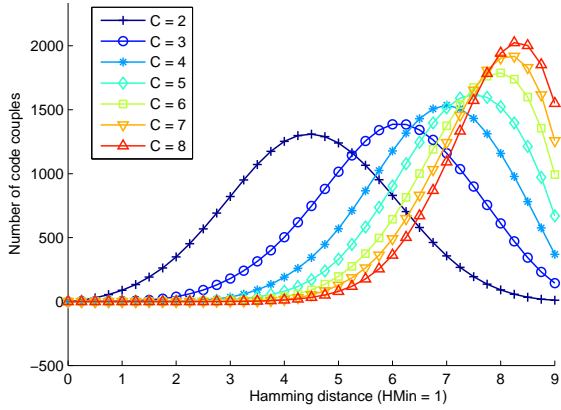


FIGURE 2 – Distributions des distances de Hamming de motifs conçus avec C symboles, $H_{Min}=1$ et $W=9$ (moyenne issue de 100 générations de matrices sous-parfaites, de taille identique, pour différentes valeurs de C).

2 Statistiques pour la conception

Dans cette section, nous présentons en premier lieu les paramètres qui agissent sur les statistiques des motifs, avant d'aborder à la section suivante la génération de motif.

2.1 Distance de Hamming moyenne

Nous reconsidérons l'idée largement répandue qu'étant donné les dimensions d'un motif, celles des fenêtres d'intérêt (les voisinages) et la valeur de la distance de Hamming minimum, utiliser moins de symboles fournit toujours la meilleure solution. Considérer un peu 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 (Fig. 2). Pour plusieurs valeurs du nombre de symboles C , pour des voisinages 3×3 (longueur du code $W = 3 \times 3 = 9$) et pour $H_{Min} = 1$, on peut observer que quand 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 avec un nombre élevé de symboles qu'avec un nombre plus faible. Cependant, on constate aussi que plus C augmente, moins le gain (sur les capacités de corrections) entre deux valeurs successives de C est important. En conclusion, l'utilisation de plus de symboles (jusqu'à une valeur qui n'induit pas de problème de discrimination des indices visuels associés au symboles) augmente le potentiel correctif embarqué dans le motif. Il serait donc intéressant de connaître l'évolution statistique du potentiel correctif du motif à concevoir. Pour conduire cette étude, nous supposons que l'évolution de la distance de Hamming moyenne représente suffisamment finement le comportement de chaque distribution.

Définition de la distance de Hamming moyenne \bar{H} .

Etant donné un ensemble S de N codes C_i de longueur W . La distance $\bar{H}(S)$ 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 :

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

$H(C_i, C_j)$ égal au nombre de positions r dans les deux codes où des symboles diffèrent :

$$H(C_i, C_j) = \sum_{r=1}^W \delta_r ; \delta_r = \begin{cases} 0 & \text{si } C_{ir} = C_{jr} \\ 1 & \text{sinon.} \end{cases} \quad (2)$$

Estimation de \bar{H} a priori. Lors de l'élaboration d'un codage basé sur les matrices (ou sous-matrices) 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 a priori dans les codes, peuvent considérer une analyse statistique à travers la valeur de \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 : Soient $A = \{A_1, \dots, A_W\}$ et $B = \{B_1, \dots, B_W\}$ deux codes de longueur W , avec $A_i, B_i \in \{0, 1, \dots, C-1\}$. 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

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

L'espérance mathématique E du nombre total de symboles identiques entre les codes A et B s'exprime par

$$\begin{aligned} E &= p(A_1 = B_1) + \dots + p(A_W = B_W) \\ &= \sum_{i=1}^W p(A_i = B_i) = W/C. \end{aligned}$$

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

Sur la Fig. 3, on observe que la définition (3) utilisée dans l'estimation a priori des valeurs de \bar{H} , pour plusieurs valeurs de H_{Min} , est très proche 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 C est très similaire à celle du potentiel correctif évoqué auparavant. En effet, la courbe s'infléchit de plus en plus au fur et à mesure que le nombre de symboles augmente.

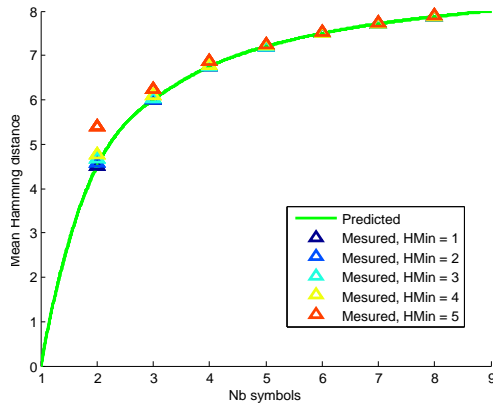


FIGURE 3 – Valeurs de \bar{H} prédites (en vert) à partir du calcul et valeurs estimées (autres couleurs) à partir de plusieurs motifs générés (moyenne sur 100 motifs), pour chaque valeur possible de H_{Min} , en fonction du nombre de symboles C ($W = 9$ ici).

3 Génération des motifs

Dans cette section, nous présentons, en plusieurs étapes, un nouvel algorithme de conception de motifs, de type "brute-force". Bien que la taille des voisinages peut être variable, la présentation ci-après considère des fenêtres d'intérêt (3×3), pour $W = 9$, ce qui permettra de comparer les performances obtenues avec d'autres travaux.

Lors du processus de construction de motif, 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. Nous proposons dans la sous-section 3.2 une technique permettant de la réduire à 1 dans le cas où $H_{Min} = 1$ et dans la sous-section 3.2 nous généralisons ceci pour des valeurs de H_{Min} plus élevées.

3.1 Méthodes existantes (sous-matrices parfaites)

Nous rappelons ici brièvement trois algorithmes de conception de motif, dus à Morano dans [12] (**M**), à Claes dans [5] (**C**) et à Maurice dans [11] (**X**), pour facilement aborder ce que nous proposons par la suite.

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 symbole 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 symbole 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 (nouvelle matrice).

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 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 incrément, les voisinages correspondants aux chiffres incrémentés sont testés. Ainsi toutes les combinaisons possibles sont examinées, du nombre composé uniquement de 0 à celui composé uniquement de $C - 1$.

Le paradigme exploration/exploitation bien connu en intelligence artificielle est appliqué dans (**X**). Grâce à l'apport d'une fenêtre de correction, il bénéficie des comportements aléatoire de (**M**) et exhaustif de (**C**). Ainsi, plus la fenêtre de correction est grande, plus le comportement de l'algorithme sera exhaustif, une plus grande zone sera explorée pour corriger les éléments conflictuels. A l'opposé, plus la taille de la fenêtre de correction est petite, plus le comportement sera aléatoire. La recherche débute par des éléments aléatoires. Cet algorithme génère en moyenne des motifs matriciels avec moins de symboles et permet de tirer parti des ressources disponibles en puissance de calculs.

Limites de ces approches Dans (**M**), la stratégie est purement aléatoire. L'algorithme ne capitalise pas les résultats du motif partiellement construit car il est réinitialisé si aucune solution pour le motif complet n'a été trouvée.

Dans (**C**), la stratégie est purement exhaustive. Si la taille des motifs ou si le nombre de symboles augmentent, l'explosion combinatoire liée à l'examen de toutes les possibilités de configuration implique que très peu de possibilités peuvent être explorées en un temps raisonnable. De plus, les chiffres de poids fort étant très peu souvent incrémentés, le symbole 0 s'y trouve sur-représenté. La distribution des symboles n'est donc pas homogène. Par ailleurs, si une solution existe pour les valeurs fixées des paramètres, l'algorithme (**M**) ne garantit pas de la trouver contrairement à (**C**). Pour (**X**), la probabilité de trouver la solution est liée à la taille de la fenêtre de correction.

3.2 Optimisations proposées

Une tâche commune à toutes ces approches concerne les tests relatifs à la distance de Hamming. A notre connaissance, ce test a toujours été réalisé dans l'espace de la matrice du motif. Ainsi, quel que soit le comportement de l'algorithme (aléatoire, exhaustif, combinaison des deux avec fenêtre de correction), la distance de Hamming entre un nouveau code et tous ceux déjà présents dans le motif doit être évaluée (H tests). Quand une matrice pseudo-aléatoire ou sous-matrice parfaite de dimensions $m \times n$ est à construire, le nombre de H tests qui doit être réalisé pour intégrer un 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 ici propose une réduction du nombre de H tests et reprend la structure générale de celle de (X) [11]. Cependant, les H tests sont réalisés dans l'espace des adresses des codes.

En théorie de l'information, il est connu que le nombre de symboles C requis pour créer N codes d'une longueur W avec une distance de Hamming minimum H_{Min} est égal à $C \geq N^{1/(W-(H_{Min}-1))}$ [19]. Il s'ensuit que pour une valeur de H_{Min} fixée, l'espace des codes C -aire, de longueur W , noté $\Omega_{C,W}$, est tel que

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

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. Une table des C^W variables booléennes est construite et l'adresse (ou l'index) de chaque variable est associée à un seul code de $\Omega_{C,W}$. Les W symboles correspondent aux coefficients A_i d'un polynôme en base C :

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

Si $\mathcal{B}(A)=False$, alors A n'a pas été inclus. Initialement, les variables booléennes sont toutes affectées avec *False*. Ainsi, pour un code A candidat, les variables booléennes pour lesquelles les codes possèdent une distance de Hamming $h < H_{Min}$ avec A sont testées. Si toutes les variables booléennes ainsi testées sont à *False*, 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 *True*.

Pour $H_{Min} = 1$, le seul code qui vérifie $h < H_{Min}$ est lui-même ($h = H_{Min} - 1 = 0$). Ceci est à comparer avec $(m \times n)/2 = 5000$ H tests pour un motif 100×100 , quand les tests sont effectués dans l'espace de la matrice.

Etant donné que 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, c'est-à-dire 16 Mo.

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 est

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

Preuve (et algorithme) : Pour chaque valeur $h \in [0, H_{Min}[$, le nombre de codes, N_C , ayant une distance de Hamming égale à h avec le code A est $N_C = N_W \times (C-1)^h$. $N_W = \frac{W!}{h!(W-h)!}$ est le nombre de combinaisons possibles de choisir h éléments parmi W . Ce sont toutes les combinaisons de symboles qui amènent

à une différence avec le code A sur h positions parmi W . Pour chaque différence avec A sur h positions, le facteur $(C-1)^h$ exprime le nombre de combinaisons possibles des symboles ; En effet, pour chaque position, il existe $C-1$ symboles qui maintiennent une différence possible avec A , il y en a donc $(C-1)^h$ sur h positions entre deux codes. Le premier terme de l'équation (6) prend en considération la valeur $h = 0$.

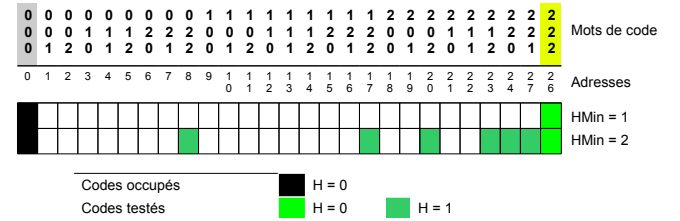


FIGURE 4 – Illustration des H tests réalisés dans l'espace des adresses des codes.

Ceci est illustré sur la figure 4 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 $N_B = 1 + \frac{3!}{(3-1)!} \cdot (3-1)^1 = 1 + 6 = 7$ au sein du tableau mono-dimensionnel des adresses, pour assurer une différence de $H_{Min} = 2$ avec les autres codes. Les 7 cellules correspondantes sont en vert (dont une en vert clair pour le code lui-même - $h = 0$, et six en vert foncé pour $h = 1$). Les autres adresses correspondent à des codes qui ont une distance de Hamming ≥ 2 avec $A = \{2, 2, 2\}$ (en blanc, et en noir pour le code $\{0, 0, 0\}$).

Stratégie de recherche partagée : L'index h apparaissant en tant qu'exposant dans (6), la méthode ci-dessus est efficace seulement pour de faibles valeurs de H_{Min} . Pour des valeurs plus élevées, nous proposons de décomposer h en faisant appel à des codes intermédiaires. Ces intermédiaires sont choisis afin de capitaliser l'effort de calculs sur les tests déjà effectués sur les codes inclus. Pour aborder ceci, nous rappelons qu'étant donné deux codes A_1 et A_2 ayant entre eux une distance de Hamming $H_{1,2}$, il existe toujours un autre code, A , ayant respectivement des distances de Hamming H_1 et H_2 avec A_1 et A_2 et telles que $H_{1,2} = H_1 + H_2$ avec $H_1 \leq H_{1,2}$ et $H_2 \leq H_{1,2}$.

Explication : Il suffit que A possède H_1 symboles de A_2 différents de ceux dans A_1 et H_2 symboles de A_1 différents de ceux dans A_2 , les symboles communs à A_1 et A_2 se retrouvant dans A . Le nombre de tests à effectuer est

$$N_B(H_{Min}) = N_B(H_1 + 1) + N_B(H_2 + 1), \quad (7)$$

avec $H_1 + H_2 = H_{Min} - 1$. Pour qu'un code intermédiaire soit atteint, il suffit d'effectuer $N_B(H_{Min})$ tests. Si à l'issue de ces tests, aucun code intermédiaire n'est obtenu, il n'y a pas de code se trouvant à une distance de Hamming

k/H _{Min}	1				2			3				4			5		
Note					Claes : N.A.							Claes : N.A.			Morano : N.A.		
Algo	Our <1s	Our Max	C.	M.	Our <1s	Our Max	M.	Our <1s	Our Max	C.	M.	Our <1s	Our Max	M.	Our <1s	Our 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

Biggest pattern, size increase : < 100% or N.A. 100% - 500% > 500%
 Biggest pattern we generate in less than 1 sec :

FIGURE 7 – Résultats comparatifs pour la génération de motif basée sur les sous-matrices parfaites (N.A. : Not Avail., M. : [Morano98] C. : [Claes08].)

ter de recourir à une invariance rotationnelle des indices. Pour toutes les reconstructions présentées ci-après, un mo-

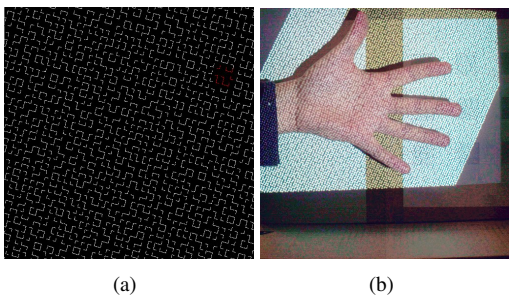


FIGURE 8 – (a) Projection d'un motif 100 × 150 avec $H_{Min} = 4$ et $C = 4$ symboles (associés à des indices visuels cunéiformes). (b) Même motif projeté sur une main.

tif de 100 × 150, avec 4 symboles et $H_{Min} = 4$ a été employé. La Fig. 9 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. En effet, grâce aux capacités d'auto-corrrections intégrées, un indice visuel totalement occulté peut néanmoins être identifié si son voisinage (et les voisinages adjacents) contiennent suffisamment d'informations pour retrouver son code.

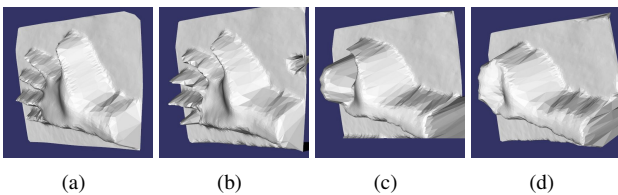


FIGURE 9 – Reconstruction en temps réel d'une main se refermant (a-d).

Sur la Fig. 10, la sensibilité du codage vis-à-vis de diverses perturbations géométriques ou spectrales est faible; les surfaces sont reconstruites sans inclure des imprécisions autres que celles dues à la résolution employée. D'ailleurs, sur la figure 11, on peut observer les différences dans la qualité de reconstruction selon la résolution du motif projeté (100 × 150 contre 50 × 50).

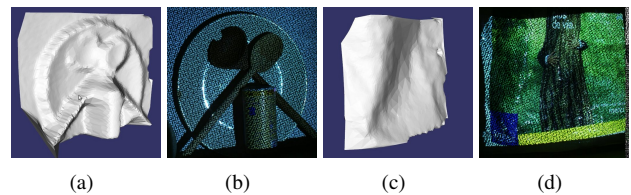


FIGURE 10 – Sensibilité du décodage vis-à-vis de diverses perturbations. (a,b) : occultations. (c,d) : perturbations spectrales.

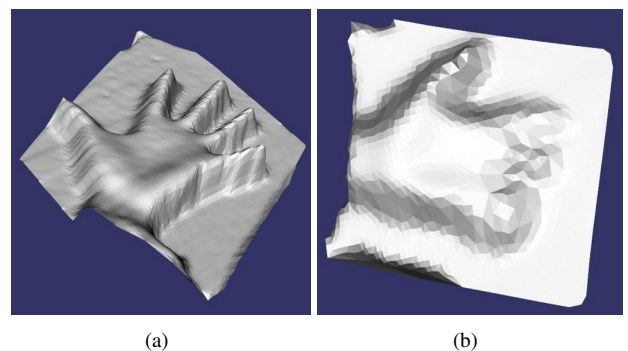


FIGURE 11 – Comparaison sur la reconstruction d'une main avec (a) une haute résolution du motif et (b) une résolution standard du motif.

4 Conclusion et perspectives

Nous avons présenté dans cet article un nouvel algorithme de conception de motif 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 oeuvre à l'aide d'algorithmes de type "brute-force", la construction informatique de motif 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. Ces tests conditionnent les performances des algorithmes de construction de motif 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 oeuvre les tests dans l'espace 1-D des adresses des codes plutôt que dans l'espace 2-D de la matrice. 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. Des reconstructions de surfaces obtenues en temps réel à partir d'une seule image démontrent le potentiel applicatif et ouvre une perspective intéressante, celle des motifs qui pourront s'adapter, en temps réel, à la scène observée.

Références

- [1] A. Adán, A. Vázquez, C. Cerrada, and S. Salamanca. Moving surface extraction based on unordered hexagonal perfect submaps projection : Applications to 3d feature tracking. *Image Vision Computing*, 27(8) :1083–1096, 2009.
- [2] C. Albitar, P. Graebing, and C. Doignon. Robust structured light coding for 3d reconstruction. In *Int'l Conf. on Computer Vision*, Rio de J., Brésil, 2007.
- [3] J. Batlle, E. Mouaddib, and J. Salvi. Recent progress in coded structured light as a technique to solve the correspondence problem : a survey. *Pattern Recognition*, 31(7) :963–982, 1998.
- [4] S. Chen, Y. 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, Avril 2007.
- [5] K. Claes. *Structured Light Adapted to Control a Robot Arm*. PhD thesis, Kath. Univ. Leuven, 2008.
- [6] N. G. de Bruijn. A combinatorial problem. In *Nederlandske Akad. van Wetenschappen*, pages 758–764, 1946.
- [7] T. Etzion. Constructions for perfect maps and pseudorandom arrays. *IEEE Int'l Trans. on Information Theory*, 34(5/1) :1308–1316, 1988.
- [8] P. M. Griffin, L. S. Narasimhan, and S. R. Yee. Generation of uniquely encoded light patterns for range data acquisition. *Pattern Rec.*, 25(6) :609–616, 1992.
- [9] H. Kawasaki, R. Furukawa, R. Sagawa, and Y. Yagi. Dynamic scene shape reconstruction using a single structured light pattern. In *Int'l Conf. on Computer Vision and Pattern Recognition*, Anchorage, 2008.
- [10] T. Koninckx and L. V. Gool. Real-time range acquisition by adaptive structured light. *IEEE Trans. on PAMI*, 28(3) :432–445, Mars 2006.
- [11] X. Maurice, P. Graebing, and C. Doignon. Real-time structured light coding for epipolar-dependent patterns. In *SPIE Photonics Europe*, volume 7724, Bruxelles, Belgique, Avril 2010.
- [12] A. Morano, C. Ozturk, R. Conn, S. Dubin, S. Zietz, and J. Nissanov. Structured light using pseudorandom codes. *IEEE Int'l Trans. on Pattern Analysis and Machine Intelligence*, 20(3) :322–327, 1998.
- [13] H. Morita, K. Yajima, and S. Sakata. Reconstruction of surfaces of 3-d objects by m-array pattern projection method. *ICCV*, 88 :468–473, 1988.
- [14] J. Pagès, J. Salvi, C. Collewet, and J. Forest. Optimised de bruijn patterns for one-shot acquisition. *Image and Vision Computing*, 23 :707–720, 2005.
- [15] K. G. Patterson. Perfect maps. *IEEE Int'l Trans. on Information Theory*, 40(3) :743–753, 1994.
- [16] R. Sagawa, Y. Ota, Y. Yagi, R. Furukawa, and N. Asada. Dense 3d reconstruction method using a single pattern for fast moving object. In *Int'l Conf. Computer Vision*, Kyoto, Japan, 2009.
- [17] J. Salvi, J. Batlle, and E. Mouaddib. A robust-coded pattern projection for dynamic 3d scene measurement. *Pattern Rec. Letters*, 19 :1055–1065, 1998.
- [18] J. Salvi, J. Pagès, and J. Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37 :827–849, 2004.
- [19] R. Singleton. Maximum distance q-nary codes. *IEEE Trans. on Information Theory*, 10 :116–118, 1964.
- [20] T. Weise, B. Leibe, and L. V. Gool. Fast 3d scanning with automatic motion compensation. In *Int'l Conf. on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA, Juin 2007.
- [21] M. Young, E. Beeson, J. Davis, S. Rusinkiewicz, and R. Ramamoorthi. View-point coded structured light. In *Int'l Conf. on Computer Vision and Pattern Recognition*, Minneapolis, MN, USA, Juin 2007.
- [22] L. Zhang, B. Curless, and S. Seitz. Rapid shape acquisition using structured light and multi-pass dynamic programming. In *Int'l Symp. on 3D Data Processing, Visualization and Transmission*, pages 24–36, 2002.