جامعة سيدي محمد بن عبد الله بفاس
UNIVERSITE SIDI MOHAMED BEN ABDALLAH DE FES
كلية العلوم ظهر المهراز فاس
FACULTE DES SCIENCES DHAR EL MEHRAZ FES

Formation Doctorale : Sciences et Technologies de l'Information et de la Communication

Discipline : Informatique

Spécialité : Informatique

Laboratoire : LABORATOIRE D'ÉLECTRONIQUE, SIGNAUX-SYSTÈMES ET INFORMATIQUE

THESE DE DOCTORAT

Présentée par

---------Tarik JAHID---------

Contributions à l'amélioration des algorithmes d'analyse des images 2D et 3D par les moments orthogonaux discrets de Meixner

Soutenue le 04/01/2020 devant le jury composé de :

Pr. Ismail BOUMHIDI	***FSDM -FES***	***Président***
Pr. Abdeljabbar CHERKAOUI	***ENSA – TANGER***	***Rapporteur***
Pr. Abdelouahid LYHYAOUI	***ENSA – TANGER***	***Rapporteur***
Pr. Abdellah AARAB	***FSDM –FES***	***Rapporteur***
Pr. Rachid EL ALAMI	***FSDM –FES***	**Examinateur**
Pr. Hassan QJIDAA	***FSDM –FES***	**Directeur de thèse**
Pr. Mhamed SAYYOURI	***ENSA - FES***	**Co-encadrant**

Année universitaire : 2018-2019

A mes parents Ahmed et Mina,

A ma Sœur Nidal, mes frères Salah et Yassine,

A mes grands Parents, à mes tantes et oncles,

A mes amis,

A Hind,

A tous ceux qui m'ont soutenu et cru en moi

Je dédie ce travail

REMERCIEMENTS

Les travaux présentés dans cette thèse ont été effectués au sein du laboratoire d'Electronique, Signaux, Systèmes et d'Informatique de la Faculté des Sciences Dhar El Mahraz de Fès; sous la direction de Monsieur le Professeur **Hassan Qjidaa**. Je tiens à lui exprimer toute ma reconnaissance pour l'encadrement de ce travail, sa grande disponibilité et l'enthousiasme dont il a fait preuve tout au long de la période d'encadrement.

J'adresse mes sincères remerciements à Monsieur le Professeur **Mhamed Sayyouri**, Co-encadrant de thèse, pour son suivi, ses conseils judicieux et ses discussions qui m'ont beaucoup aidé au cours de mes recherches.

Je remercie Monsieur **Ismail BOUMHIDI**, Professeur à la Faculté des Sciences Dhar El Mahraz, de Fès, pour avoir accepté de me faire l'honneur de présider le jury de ma soutenance.

Mes sincères remerciements à Monsieur **Abdeljabbar CHERKAOUI**, Professeur à l'Ecole Nationale des Sciences Appliquées de Tanger, et Monsieur **Abdelouahid LYHYAOUI**, Professeur à l'Ecole Nationale des Sciences Appliquées de Tanger, ainsi que Monsieur **Abdellah AARAB**, Professeur à la Faculté des Sciences Dhar El Mahraz de Fès, d'avoir pris le temps de juger ce travail et de m'avoir fait l'honneur d'être rapporteurs de ma thèse.

Je tiens également à remercier Monsieur **Rachid EL ALAMI**, Professeur à la Faculté des Sciences Dhar El Mahraz, pour avoir accepté d'examiner ce travail et de faire partie de ce jury.

Mes remerciements vont aussi à tous ceux qui m'ont soutenu ou qui, d'une manière ou d'une autre, ont contribué à l'élaboration de ce travail.

Je tiens aussi à exprimer ma profonde sympathie à toutes les personnes qui composent le laboratoire LESSI.

PUBLICATIONS

Articles:

[1] Jahid, T., Hmimid, A., Karmouni, H., Qjidaa, H., Rezzouk, A. : "Image analysis by Meixner moments and a digital filter" , Multimedia Tools and Applications, Springer ; 2018

[2] Jahid, T., Karmouni, H., Hmimid, A., Sayyouri, M., Qjidaa, H. : "Fast computation of Charlier moments and its inverses using Clenshaw's recurrence formula for image analysis" , Multimedia Tools and Applications, Springer; 2019

[3] Jahid, T., Karmouni, H., Sayyouri, M., Hmimid, A., Qjidaa, H. : "Fast Algorithm of 3D Discrete Image Orthogonal Moments Computation Based on 3D Cuboid" , Journal of Mathematical Imaging and Vision , Springer ; 2019

[4] Karmouni, H., Hmimid, A., Jahid, T., Qjidaa, H., Rezzouk, A. : "Fast and Stable Computation of the Charlier Moments and Their Inverses Using Digital Filters and Image Block Representation" , Circuits, Systems, and Signal Processing,Springer ; 2018

[5] Karmouni, H., Jahid, T., Hmimid, A., Sayyouri, M., Qjidaa, H. : "Fast computation of inverse Meixner moments transform using Clenshaw's formula", Multimedia Tools and Applications, Springer ; 2019

[6] Karmouni, H., Jahid, T., Sayyouri, M., El Alami, R., Qjidaa, H. :"Fast 3D image reconstruction by cuboids and 3D Charlier's moments" , Journal of Real-Time Image Processing, Springer ;2019

[7] Karmouni, H., Jahid, T., Sayyouri, M., Hmimid, A., Qjidaa, H. :"Fast Reconstruction of 3D Images Using Charlier Discrete Orthogonal Moments Circuits", Systems and Signal Processing, Springer;2019

Chapitres :

[8] Karmouni, H., Jahid, T., Sayyouri, M., El-affar, A., Qjidaa, H.:"Fast and stable computation of the Tchebichef's moments using image block representation and Clenshaw's formula" , Advances in Intelligent Systems and Computing, Springer ; 2019

[9] Karmouni, H., Jahid, T., Sayyouri, M., El-Affar, A., Qjidaa, H. : "Image analysis by Hahn moments and a digital filter ", Advances in Intelligent Systems and Computing, Springer ; 2019

Communications:

[10] Jahid, T., Karmouni, H., Hmimid, A., Sayyouri, M., Qjidaa, H., : “Image moments and reconstruction by Krawtchouk via Clenshaw's reccurence formula”, Proceedings of 2017 International Conference on Electrical and Information Technologies, ICEIT 2017;IEEE; 2018

[11] Hicham Karmouni ; Tarik Jahid ; Imad El Affar ; Mhamed Sayyouri ; Abdeslam Hmimid ; Hassan Qjidaa ; Abdellah Rezzouk : “Image analysis using separable Krawtchouk-Tchebichef's moments” ,2017 International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), IEEE ; 2017

[12] Hicham Karmouni ; Tarik Jahid ; Zouhir Lakhili ; Abdeslam Hmimid ; Mhamed Sayyouri ; Hassan Qjidaa ; Abdellah Rezzouk : ”Image reconstruction by Krawtchouk moments via digital filter”, 2017 Intelligent Systems and Computer Vision (ISCV), IEEE; 2017

[13] Karmouni, H., Jahid, T., Sayyouri, M., El-affar, A., Qjidaa, H.:”Fast and stable computation of the Tchebichef’s moments using image block representation and Clenshaw’s formula” , Advances in Intelligent Systems and Computing, Springer ; 2019

[14] Karmouni, H., Jahid, T., Sayyouri, M., El-Affar, A., Qjidaa, H. : “Image analysis by Hahn moments and a digital filter “, Advances in Intelligent Systems and Computing, Springer ; 2019

TABLE DES MATIERES

Remerciements 2

Publications….. 3

Table des matières 5

Liste des figures..... 9

Liste des tableaux 12

Liste des abréviations, des sigles et des acronymes 13

Résumé:……..... 16

Abstract :…… 17

Introduction GENERALE 19

Chapitre 1 : Généralité sur les polynômes et les moments continues et discrets 23

1. Généralités sur les Polynômes orthogonaux à une seule variable..... 24
   1.1. Notion d'orthogonalité 25
   1.2. Procédé d'orthogonalisation 25
   1.3. Formules de récurrence 26
   1.3.1. Base orthogonale 26
   1.3.2. Base orthonormée..... 27
   1.4. Formule Christoffel-Darboux..... 27
   1.5. Types de polynômes orthogonaux 28
   1.5.1. Polynômes orthogonaux d'une variable continue..... 28
   1.5.2. Polynômes orthogonaux d'une variable discrète 28
   1.6. Série hypergéométrique de base..... 29
   1.6.1. Propriétés élémentaires des séries hypergéométriques 30
   1.6.2. Opérations sur les séries hypergéométriques 31
2. Polynômes et moments continues 32

2.1. Polynômes orthogonaux d'une variable continues 32
2.1.1. Les polynômes de Jacobi 33
2.1.2. Les polynômes de Legendre 36
2.1.3. Les polynômes de Tschebyshev 37
2.1.4. Les polynômes de Gegenbauer ou ultrasphériques 39
2.1.5. Les polynômes de Laguerre 40
2.1.6. Les polynômes d'Hermite 41
2.1.7. Les polynômes de Bessel 43
2.2. Les moments orthogonaux continus 45
2.2.1. Les moments orthogonaux continus sur un rectangle 46
2.2.2. Les moments de Legendre 47
2.2.3. Les moments de Chebyshev 48
2.2.4. Les moments de Gegenbauer 50
2.2.5. Les moments de Gauss-Hermite 52
2.2.6. Les moments continus orthogonaux sur un disque 53
2.2.7. Les moments de Zernike 54
2.2.8. Les moments orthogonaux de Fourier-Mellin 57
3. Polynômes et moments discrets 59
3.1. Polynômes orthogonaux d'une variable discrète 59
3.1.1. Forme générale des polynômes orthogonaux discrets 59
3.1.2. Relation de récurrence par rapport à l'ordre n 60
3.1.3. Propriété d'orthogonalité des polynômes orthogonaux discrets 60
3.1.4. Domaines d'orthogonalité des polynômes orthogonaux discrets 61
3.1.5. Relation de récurrence des polynômes orthonormés discrets 62
3.1.6. Les polynômes de Tchebichef 63
3.1.7. Les polynômes de Krawtchouk 64
3.1.8. Les polynômes de Hahn 66
3.1.9. Les polynômes de Meixner 67
3.1.10. Les polynômes de Charlier 68
3.1.1. Relation de récurrence par rapport à la variable x 69
3.2. Les moments orthogonaux discrets 71
3.2.1. Les moments de Tchebichef 72
3.2.2. Les moments de Krawtchouk 73
3.2.3. Les moments de Racah 74
3.2.4. Les moments de Charlier 76
3.2.5. Les moments de Hahn 77
4. Conclusion 78

Chapitre 2: Calcul Rapide des moments de meixner et leurs inverses pour des images 2D 79

1. Introduction 79
2. Les polynômes et moments de Meixner 80
2.1. Les polynômes de Meixner 80
2.1.1. Relation de récurrence par n 82
2.1.2. Relation de récurrence par x 83

2.2. Les moments de Meixner 84
2.3. La transformée inverse 84
3. La formule de récurrence de Clenshaw 85
3.1. Calcul des moments de Meixner via la relation de récurrence de Clenshaw 86
3.2. Reconstruction d'image 2D par les moments de Meixner via la formule de Clenshaw… 89
4. Temps de calcul des moments de Meixner 93
5. Temps de calcul de la transformée inverse des moments de Meixner 96
6. Reconstruction d'image 2D par les moments de Meixner 99
7. Conclusion 103

Chapitre 3 : Calcul des moments de Meixner et leurs inverse par la méthode des blocs via les filtres numériques 104

1. Introduction 104
2. Les moments de Meixners 106
3. Calcul rapide des moments de Meixner par le filtre numérique 108
3.1. Les sorties du filtre numérique en cascade 109
3.2. Calcul des moments de Meixner 110
3.3. La transformée inverse des moments de Meixner pour une image 2D 110
3.4. Méthode de reconstruction d'image 2D par blocs fixes 111
4. Calcul des moments de Meixners et leurs inverse par la méthode IBR-ISR 114
4.1. Image binaire 114
4.2. Image à niveaux de gris 115
5. Simulations et résultats 119
5.1. Le temps de calcul des moments de Meixner 119
5.2. La reconstruction d'image 2D par les méthodes proposées 122
6. Conclusion 125

Chapitre 4 : Algorithme rapide de calcul des moments orthogonaux discrets sur des images 3D basé sur la ICR 126

1. Introduction 126
2. Moments et polynômes orthogonaux discrets : forme générale 128
3. Représentation d'image 3D par cuboïdes (ICR) 132
4. Calcul rapide des moments de Meixner et leurs inverses via l'ICR 135
4.1. Calcul des moments de Meixners et leurs inverses : méthode classique 135
4.2. Calcul rapide des moments de Meixner et leurs inverses : méthode ICR 136
4.3. Calcul rapide des moments de Meixner et leurs inverses : Méthode matricielle 139
5. Moments orthogonaux invariants à travers les cuboïdes 146
5.1. Moments géométriques invariants 3D 148
5.2. Moments Invariant de Meixner 3D 151
5.3. Calcul rapide des moments invariants de Meixner 3D 155
6. Simulations et résultats : Moments et Reconstruction d'images 3D 158
6.1. Extraction des cuboïdes sur une image 3D discrète 159
6.2. Temps de calcul des moments de Meixner et leurs inverses pour une image 3D 161

6.3. Temps de calcul des moments orthogonaux discrets et leurs inverse pour d'autres bases 162
6.4. Reconstruction d'image 3D par les moments de Meixner 163
6.5. Reconstruction d'im age 3D par les moments orthogonaux discrets : méthode proposé 166
7. Simulations et résultats : Classification des mages 3D 170
7.1. Extraction des cuboïdes 170
7.2. Calcul des moments invariant orthogonaux de Meixner 172
7.3. Invariance TRS : Taranslation, Rotation, Scale 174
7.4. Classification 180
8. Conclusion 183

Conclusions et perspectives 185

Références …. 187

LISTE DES FIGURES

Figure (I.1): famille des moments 24
Figure (I.2): Le schéma Askey pour les polynômes hypergéométriques orthogonaux, avec les relations de la limite indiquées entre les polynômes. 29
Figure (I.3): Courbes des polynômes orthogonaux de Jacobi pour $\alpha = 0.5 \; et \; \beta = 0.25$ *et* $\alpha = 1.75 \; et \; \beta = 1.25$ *pour les six premiers ordres.* 35
Figure (I.4) : Courbes des polynômes orthonormés de Legendre pour les six premiers ordres. 37
Figure (I.5) : Courbes des polynômes orthonormés de Chebyshev de première espèce et de deuxième espèce pour les six premiers ordres. 38
Figure (I.6): Courbes des polynômes orthogonaux de Gegenbauer ($\lambda = -0.5$*) et (*$\lambda = 0.75$*) pour les six premiers ordres.* 40
Figure (I.7): Courbes des polynômes orthonormés de Laguerre pour ($\alpha = 0$*) et (*$\alpha = 0.75$*) pour les six premiers ordres.* 41
Figure (I.8): Courbes des polynômes orthogonaux d'Hermite pour les six premiers ordres. 43
Figure (I.9): Courbes des polynômes orthogonaux de Bessel de première espèce et deuxième espèce pour les six premiers ordres. 45
FIGURE (I.10): Courbes des polynômes radiaux de Zernike pour les six premiers ordres. 57
FIGURE (I.11): Courbes des polynômes orthogonaux de Fourier-Mellin pour les six premiers ordres. 58
Figure (I.12): Courbes des polynômes orthonormés de Tchebichef pour les six premiers ordres 64
Figure (I.13): Courbes des polynômes orthonormés de Krawtchouk pour $p = 0.2, \; p = 0.5, \; et \; p = 0.9$ *pour les six premiers ordres.* 66
Figure (I.14): Courbes des polynômes orthonormés de Hahn pour ($a = b = 10$*) et (*$a = 30 \; ; \; b = 10$*) pour les six premiers ordres.* 67
Figure (I.15): Courbes des polynômes orthonormés de Meixner pour $\mu = 0.25, \beta = 40 \; et \; \mu = 0.5, \beta = 60$ *pour les six premiers ordres.* 68
Figure (I.16): Courbes des polynômes orthonormés de Charlier pour $a_1 = 40 \; et \; a_1 = 80$ *pour les six premiers ordres.* 69
Figure (II.1): Ensemble d'images binaires de dimension 128×128 *pixels: Tree, Children et Deer* 95
Figure (II.2): Temps moyens (CPU) nécessaire pour le calcul des moments de Meixner de différents ordres pour des images binaires 95
Figure (II.3): Ensemble d'images de niveaux de gris de dimensions 150×150 *pixels: House, Pirate et Lake* 95
Figure (II.4): Temps moyens (CPU) nécessaire pour le calcul des moments de Meixner de différents ordres pour des images de niveaux de gris 96
Figure (II.5): Ensemble d'images binaires de dimensions 128×128 *pixels: Bird, Device et Pocket* 97
Figure (II.6): Temps moyens (CPU) nécessaire pour le calcul des moments de Meixner inverses de différents ordres pour des images binaires 98

Figure (II.7): Ensemble d'images de niveaux de gris de dimensions 150×150 *pixels: House, Pirate et Lake* 98
Figure (II.8): Temps moyens (CPU) nécessaire pour le calcul des moments de Meixner inverses de différents ordres pour des images de niveaux de gris 99
Figure (II.9): image binaire « Bird »: (a) noise-free image (sans bruit), (b) noisy image: Salt & Peppers (bruit sel et poivre), (c) noisy image: white Gaussian (bruit Gaussian blanc) 100
Figure (II.10): image de niveaux de gris « Pepper »: (a) noise-free image (sans bruit), (b) noisy image: Salt & Peppers (bruit sel et poivre), (c) noisy image: white Gaussian (bruit Gaussian blanc) 101
Figure (II.11): MSE de l'image binaire reconstruite « bird ». 101
Figure (II.12): MSE de l'image reconstruite en niveaux de gris « Pepper » 101
Figure (II.13): Reconstruction d'image binaire sans et avec bruit pour des ordres différents. 102
Figure (II.14): Reconstruction d'image de niveaux de gris sans et avec bruit pour des ordres différents 102
Figure (III.1): Structure de filtre numérique pour généré les sorties désirés : (a) y_n *de 1D pour (n) ordres. (b)* $y_{n,m}$ *de 2D pour (n+m) ordre.* 109
Figure (III.2): Illustration des blocs fixes : (a) Division de l'image originale en Nb sous-images, (b) extractions des moments de chaque bloc, et (c) reconstruction de l'image en fusionnant les Nb blocks à partir de (b). 111
(c) 112
Figure (III.3): Reconstruction d'image par la méthode des blocs fixes 114
Figure (III.4): Extraction des blocs par l'algorithme IBR 115
Figure (III.5): Extraction des blocs en combinant les algorithmes ISR-IBR. 116
Figure (III.6): Extraction of blocks using ISR algorithm 117
Figure (III.7): une fonction discrète f(x,y) de dimensions 4x4 117
Figure (III.8): Blocs transformé par le filtre digital 118
Figure (III.9): Outputs of filtred blocks 118
Figure (III.10): Ensemble d'images binaires de dimensions 256x256 (a) Horse, (b) Bird, (c) Elephant, (d) Bat and (e) Hammer 120
Figure (III.11): Représentation d'image par blocs (IBR): (a) Horse, (b) Bird, (c) Elephant, (d) Bat and (e) Hammer 121
Figure (III.12): Partition d'image binaire en blocs fixes de tailles 8x8:Nb=256 (a) Horse, (b) Bird, (c) Elephant, (d) Bat and (e) Hammer 121
Figure (III.13): Ensemble d'image de niveaux de gris de domensions 256x256 (a) Lena, (b) Boat, (c) Mandrill, (d) Pepper and (e) Elaine 121
Figure (III.14): Représentation par slice-bloc (ISR-IBR): (a) Lena, (b) Boat, (c) Mandrill, (d) Pepper and (e) Elaine 121
Figure (III.15): partition en blocs fixes 8x8:Nb=256:(a) Lena, (b) Boat, (c) Mandrill, (d) Pepper and (e) Elaine 122
Figure (III.16): Reconstruction d'un bloc. 123
Figure (III.17): (a) « Barbara » image de niveaux de gris de dimensions 256x256, (b) noisy by Salt & Peppers (bruité en sel et poivre) and (c) noisy image by white Gaussian (bruit Gaussian blanc) 124
Figure (IV.1): Représentation d'image 3D discrète dans l'espace 133
Figure (IV.2): Image "Ant" de la base McGill 134
Figure (IV.3): plusieurs angles de vues de la représentation de l'image "Ant" par cuboïdes. 134
Figure (IV.4): Ensmble d'image 3D: Cup, Chair, Human, Octopuse and Spectacles de la base McGILL. 160
Figure (IV.5): (a) Temps moyen pour le calcul des moments de Meixner pour des images 3D (b) Temps moyens pour calculer leurs inverses. en utilisant la méthode récursive et la méthode proposé Directe et par produit matriciel 161

Figure (IV.6): (a) Temps moyen de calcul des moments de Tchebichef, Krawtchouk, Hahn, Charlier and Meixner pour une image 3D. (b) Temps moyens de calcul de leurs inverses respectivement, en utilisant la méthode des cuboïdes 162
Figure (IV.7): Mean squared error « MSE » et Peak Signal-To-Noise Ratio « PNSR » pour l'image reconstruite "Cup" Par la méthode proposé directe et avec produit matriciel 164
Figure (IV.8): l'image reconstruite 3D " Cup " pour les ordres 20, 30, 40, 50, 80, respectivement par la récursive et la méthode proposé 165
Figure (IV.9) : Structural Similarity « SSIM » pour image 3D "Cup" Par la méthode des cuboïdes 166
Figure (IV.10) (a) Mean squared error « MSE » and (b) Peak Signal-To-Noise Ratio « PNSR » pour l'image reconstruite « Cup » par les moments de Tchebichef, Krawtchouk, Hahn, Charlier and Meixner. 167
Figure (IV.11): Image 3D reconstruite " Ant " pour les ordres 10, 40, 70,réspectivement par les moments de Tchebichef, Krawtchouk, Hahn, Charlier et Meixner. 169
Figure (IV.12): Noise-free and noisy "Airplane image". 169
Figure (IV.13) Mean squared error « MSE » for the noise-free and noisy "Airplane image" by proposed method. 169
Figue (IV.14): Représentation d'mage par cuboïde (ICR), pour des images binaires 3D " Dog " et " Cup ".... 171
Figure (IV.15): Représentation d'image 3D par cuboïde (ICR), pour des images multi-niveaux 3D " MRI 1" et " MRI 2" 172
Figure (IV.16): Temps de calcul pour un ensemble d'images 3D, (a) : binaire, et (b) : multi-niveaux, en utilisant les moments invariant de : Meixner (CMIM), Tchebichef (CTIM), Krawtchouk (CKIM), Hahn classique (CHIM), et les moments invariants de Meixner proposés (PMIM). 174
Figure (IV.17): L'image binaire 3D: (a) "Table", et l'image multi-niveaux 3D (b) "MRI 3" de taille 300×300×300 voxels. 174
Figure (IV. 18) : Erreurs relatives pour l'image binaire 3D «Table» et l'image multi-niveaux 3D «MRI 3» entre les images originales et les versions à changement d'échelle en utilisant le PMMI proposé. 175
Figure (IV.19): Erreurs relatives pour l'image binaire 3D «Table» et l'image multi-niveaux 3D «MRI 3» entre les images originales et les versions à translation en utilisant le PMMI proposé. 175
Figure (IV.20): Erreurs relatives pour l'image binaire 3D «Table» et l'image multi-niveaux 3D «MRI 3» entre les images originales et les versions à rotation en utilisant le PMMI proposé. 176
Figure (IV.21): Erreurs relatives pour l'image binaire 3D «Table» et l'image multi-niveaux 3D «MRI 3» entre les images originales et les versions à les images corrompues en utilisant le PMMI proposé. 176
Figure (IV.22): Image transformée"Teddy-bears" utilisé dans cette expérience. L'image est montrée avec leurs transformations. 177
Figure (IV.23): Image transformée" MRI 3 " utilisé dans cette expérience. L'image est montrée avec leurs transformations. 178
Figure (IV.24): Collection de 9 images sélectionnées dans la base de données PSB et utilisées comme ensemble d'apprentissage 182

LISTE DES TABLEAUX

Tableau (I.1): Les données de base pour les polynômes orthogonaux discrets de Tchebichef, Krawtchouk, Hahn, Meixner et Charlier. 62
Tableau (I.2): Les données de la relation de récurrence à trois termes des polynômes orthogonaux discrets. ... 63
Tableau (I.3) : Les valeurs initiales de la relation de récurrence à trois termes par rapport à la variable x pour les polynômes de Tchebichef, Krawtchouk, Hahn, Meixner et Charlier. 70
Tableau (II.1) : Complexité de calcul des moment1D de Meixner MM_n *de l'ordre* n 88
Tableau (II.2) Compléxité de calcul des moments 2D de Meixner MM_n *de l'ordre* $(n+m)$ 89
Tableau (II.3) : Compléxité de calcul des moments inverse $1D$ *Meixner* 91
Tableau (II.4): Complexité du calcul des moments inverse $2D$ *de Meixner* 93
Tableau (II.5): Temps moyens et rapport ETIR pour le calcul des moments d'ordres différents pour des images binaires 95
Tableau (II.6) : Temps moyens et rapport ETIR pour le calcul des moments d'ordres différents pour des images de niveaux de gris 95
Tableau (II.7) : Temps moyens et rapport ETIR pour le calcul des moments inverses d'ordres différents pour des images binaires 97
Tableau (II.8): Temps moyens et rapport ETIR pour le calcul des moments inverses d'ordres différents pour des images de niveaux de gris 98
Tableau (III.1): Temps moyens pour le calcul des moments Meixner pour des images binaires 121
Tableau (III.2): Temps moyens de calcul des moments de Meixner pour des images de niveaux de gris. 122
Tableau (III.3): Images reconstruites et leurs Mean squared error (MSE) 124
Tableau (III.4): Images reconstruites et leurs Mean squared error (MSE). 125
Tableau (IV.1) : Polynômes Orthogonal généralisés 130
Tableau (IV.2):les termes initiaux 130
Tableau (IV.3): nombre de cuboïdes extraits par image. 159
Tableau (IV.4): Temps de calcul moyen global et ETIR (%) des moments invariants de Meixner selon la méthode proposée et classique pour les images binaires 3D et multi-niveaux. 173
Tableau (IV.5): Une comparaison des moments des invariants de Meixner 3D classiques et proposés d'image 3D "Teddy-bears". 179
Tableau (IV.6): Une comparaison des moments des invariants de Meixner 3D classiques et proposés d'image 3D "MRI 3". 179
Tableau (IV.7): Résultats de la classification utilisant le 1-NN basé sur la distance euclidienne d_1 182
Tableau (IV.8): Résultats de classification utilisant 1-NN basé sur la distance des blocs de ville d2. 182
Tableau (IV.9): Résultats de classification utilisant le 1-NN basé sur la distance de corrélation d3. 183

LISTE DES ABREVIATIONS, DES SIGLES ET DES ACRONYMES

Symboles	Significations
σ	Polynôme de 2$^{\text{iéme}}$ degré.
τ	Polynôme de 1$^{\text{iéme}}$ degré.
λ_n	Une constante appropriée.
Δ	L'opérateur de différence en avance.
∇	L'opérateur de différence en arrière.
$P_n(x)$	Le $n^{ème}$ polynôme d'une variable discrète x .
$C_n^{a_1}(x)$	Les polynômes orthogonaux discrets de Charlier.
$H_n^{(a,b)}(x,N)$	Les polynômes orthogonaux discrets de Hahn.
$t_n(x,N)$	Les polynômes orthogonaux discrets de Tchebichef.
$K_n^p(x,N)$	Les polynômes orthogonaux discrets de Krawtchouk.
$M_n^{(\gamma,\mu)}(x)$	Les polynômes orthogonaux discrets de Meixner.
B_n	Le facteur de normalisation.
$\omega_n(x)$	La fonction de poids de l'ordre *n*.
$\omega(x)$	La fonction de poids de l'ordre *n=0*.
∇^n	L'opérateur de différence en arrière de l'ordre *n*.
${}_rF_s$	La série hypergéométrique généralisée.
$(a)_k$	Le symbole de Pochhammer de *a*.
Γ	La fonction gamma.
$\langle a \rangle_k$	La factorielle décroissante.
$S(k,i)$	Le nombre de Stirling.
$\rho(n)$	La fonction norme carrée.
δ_{nm}	La fonction de Dirac.
MM_n	Les moments de Meixner d'ordre (n)
MM_{nm}	Les moments de Meixner d'ordre ($n+m$)
MM_{nmk}	Les moments de Meixner d'ordre ($n+m+k$)
$f(x,y)$	Fonction d'une image 2D

$f(x,y,z)$	Fonction d'une image 3D
MPEG-7	La base d'image MPEG-7 CE-shape-1 Part B.
McGill	La base de données 3D Shape Benchmark de McGill.
ETIR%	Le taux d'amélioration de temps d'exécution.
MSE	L'erreur quadratique moyenne pour une image.
$PSNR$	Le rapport Peak Signal-To-Noise Ratio.
$H(z)$	La fonction de transfert d'ordre $n=0$
$H_n(z)$	La fonction de transfert d'ordre n
y_n	La sortie des filtres numériques 1D.
y_{nm}	La sortie des filtres numériques 2D.
y_{nmk}	La sortie des filtres numériques 3D.
ICR	La représentation d'image 3D par cuboïdes.
Cub	Le Cuboïde d'une image 3D
MC	Les moments de Charlier
MH	Les moments de Hahn
MT	Les moments de Tchebichef
MK	Les moments de Krawtchouk
MM	Les moments de Meixner
GM_{nmk}	Les moments géométriques 3D (n+m+k).
μ_{nmk}	Les moments géométriques centraux 3D (n+m+k).
GMI_{nmk}	Les moments géométriques invariants 3D (n+m+k).
$\alpha_{n,i}^{a_1}$	Les coefficients des monômes du polynôme de Charlier.
$PGMI_{nmk}$	Les moments géométriques invariants proposés 3D (n+m+k).
$PGMI_{nmk}^{i}$	Les moments géométriques invariants proposés d'intensité i 3D (n+m+k).
$PGMI_{nmk}^{cub_j^{f_i}}$	Les moments géométriques invariants proposés d'intensité i de chaque cuboïde j 3D (n+m+k)
MMI_{nmk}	Les moments invariants de Meixner 3D d'ordre (n+m+k)
MIC	Les moments invariants de Charlier
MIH	Les moments invariants de Hahn
MIT	Les moments invariants de Tchebichef
MIK	Les moments invariants de Krawtchouk
MIM	Les moments invariants de Meixner
V	Le vecteur de primitives.
TV	Le vecteur de translation.
$E(f,g)$	L'erreur relative entre f et g .

d_1	La distance Euclidienne
d_2	La distance de bloc de ville
d_3	La distance de corrélation
η	La précision de la reconnaissance.

RESUME :

Dans cet ouvrage, nous proposons nos contributions pour le calcul des moments orthogonaux discrets en générale et ceux de Meixner en particulier, pour des images discrètes bidimensionnelles (2D) et tridimensionnelles (3D), ainsi que leurs reconstruction et classification par ces derniers. Nos contributions sont étalées, en trois parties. La première contribution est l'accélération des calculs des moments et leurs inverses pour des images 2D grâce à l'introduction de la formule de récurrence de Clenshaw, qui a un apport sur la rapidité mais n'améliore en rien la reconstruction des images 2D. La deuxième contribution est la répartition des images en slices et blocs par l'algorithme proposé (ISR-IBR), ce qui permet d'améliorer la qualité de reconstruction pour les images 2D tout en permettant un calcul rapide des moments et leurs inverses par rapport aux méthodes classiques. La troisième contribution, qui n'est d'autre que l'aboutissement d'un tel cheminement, a été dédié aux images 3D, par la répartition des images 3D en cuboïdes homogènes de même intensité (ICR) nous avons pu réduire énormément le temps de calcul des moments orthogonaux discrets et leurs inverses pour les images discrètes 3D, mais pas que, nous avons aussi amélioré la reconstruction de ces dernières et prouver qu'ils ont un réel apport à l'amélioration de la classification de ces dernières.

ABSTRACT :

In this book, we propose our contributions for the computation of discrete orthogonal moments in general and those of Meixner in particular, for bi-dimensional (2D) and three-dimensional (3D) discrete images, as well as their reconstruction and classification. Our contributions are spread out in three parts. The first contribution is the acceleration of the calculations of the moments and their inverses for 2D images with the introduction of the formula of Clenshaw's recurrence, which only contributes to accelerate the computation but does not improve the reconstruction of 2D images. The second contribution is the images repartition in slices and blocks by the proposed algorithm (ISR-IBR), which makes it possible to improve the quality of reconstruction for the 2D images while allowing a fast computing of the moments and their inverses with respect to classical methods. The third contribution, which is none other than the achievement of such a path, was dedicated to 3D images, by the repartition of 3D images in homogeneous cuboids of the same intensity (ICR), we were able to greatly reduce the computation time of discrete orthogonal moments and their inverse for 3D discrete images, but not only, we also improve the reconstruction of these and prove that they have a real contribution to the improvement of the classification 3D images.

INTRODUCTION GENERALE

Contexte :

Dans un monde en ébullition technologique, l'imagerie est au croisé de chemin des différentes disciplines scientifiques et techniques, du immensément grand au infiniment petit les différentes disciplines ont recours à l'imagerie pour comprendre le monde et l'univers. Par la multitude de l'information collectée par une simple image, les chercheurs ainsi que les simples utilisateurs comprennent certains phénomènes, vu que l'information visuelle est la plus descriptif souvent. Dans cette optique, certains chercheurs ont dédié leurs efforts à étudier l'image et développer les techniques qui peuvent extraire des informations précises pour décrire un phénomène observé dans l'image, aussi afin d'automatiser la compréhension des informations contenus par ces dernières et les comparer à d'autres collectés ailleurs, pour une compréhension plus global et plus précise. Dans ce contexte global, une technique particulière a attiré notre attention, celle « des moments d'image », qui représentent certains descriptifs caractérisant de façon cohérente une image et permettant de la reconstruire à partir de ces derniers avec une certaine précision mais aussi de la classifier parmi d'autres.

Les moments sont des grandeurs scalaires utilisées pour caractériser une fonction et capturer ses caractéristiques importantes. Ils ont été largement utilisés dans les statistiques pour la description de la forme d'une fonction de densité de probabilité et dans la mécanique classique de corps rigide pour mesurer la distribution de masse d'un corps. Du point de vue mathématique, les moments sont des «projections» d'une fonction sur une base polynomiale (comme, la transformée de Fourier est une projection sur une base de fonctions harmoniques). [Flusser, 1993]

La première étude concernant les moments invariants, a été faite par le mathématicien allemand David Hilbert au 19ème siècle [Hilbert, 1993], dans l'étude de la théorie des invariants algébriques. Puis ont été fondamentalement introduits par Hu [Hu, 1962], pour la reconnaissance des formes, qui a exploité les résultats de la théorie des invariants algébriques et a dérivé ses fameux sept invariants à la rotation aux objets bidimensionnels, ils seront nommé plus tard les moments géométrique de Hu. Pour pouvoir

atteindre des ordres plus importants, les moments rotationnels ont été présentés, puis les moments complexes ont fait leurs apparitions comme une méthode simple pour dériver les moments invariants. Ces derniers peuvent être exprimés en moments géométrique et vice-versa, ils ont été appelés communément des moments non-orthogonaux.

La description des images avec des moments signifie qu'on utilise les propriétés globales de l'image plutôt que ses propriétés locales. Malgré la simplicité des moments géométrique, ils présentaient un défaut majeur, qui est une forte redondance d'information, et ce qui rendait la reconstruction des images à partir de ces derniers considérablement difficile et inefficace. Ainsi, Teague [Teague, 1980] a proposé l'utilisation des moments orthogonaux (en projetant l'image sur une base orthogonale), en choisissant les polynômes de Zernike et de Legendre dans un domaine continue, la redondance d'information a été minimisée remarquablement, en plus de la possibilité d'obtenir des moments d'ordre très important et l'invariance à la rotation prouvée pour les moments de Zernike. Avec l'évolution technologique, d'autres problèmes ont été abordés, dont principalement l'approximation des intégrales sachant que les polynômes utilisés ont été définis sur des domaines continues alors que l'image été issue d'un domaine discrets, plusieurs solution ont été alors proposés [Mukundan, 1998] [Liao Pawlak, 1996]. Cela dit, les approximations proposés n'étaient pas efficaces et encore couteuse en temps de calcul, c'est alors que Mukundan a introduit les moments orthogonaux discrets de Tchebyshev comme solution à la discrétisation des moments. Plusieurs ont suivi, en introduisant d'autres bases dans le domaine discret Krawtchouk [Yap et al., 2003], Hahn [Yap et al, 2007; Sayyouri et al, 2012 ; Sayyouri et al, 2013], Charlier [Zhu et al., 2010; Sayyouri et al, 2012b], Racah [Zhu et al., 2007a], Dual Hahn [Zhu et al., 2007c, Karakasis et al., 2013] et Meixner [Zhu et al., 2010].

L'utilisation des polynômes discrets orthogonaux comme fonctions de base pour le calcul des moments de l'image, élimine le besoin de l'approximation numérique, et satisfait exactement la propriété de l'orthogonalité dans le domaine discret de l'espace des coordonnées de l'image. Cette propriété rend les moments orthogonaux discrets supérieurs aux moments orthogonaux continus conventionnels dans le sens de la capacité de la représentation de l'image. Les moments orthogonaux possèdent les propriétés de robustesse contre le bruit et une redondance nulle [Teh & Chin, 1988]. De plus, ils ne nécessitent aucune information à priori. Ils sont aussi insensibles aux dégradations locales des images traitées du

fait de leur traitement global. Aussi, la reconstruction d'image à partir de ses moments orthogonaux été devenu possible grâce à la transformée inverse des moments orthogonaux.

Problématiques et résultats attendus :

Les différents types des moments ont été classifiés par [Shu, 2007], en deux familles, les moments non-orthogonaux et les moments orthogonaux, des relations entre les différents moments ont été aussi prouvé. Néanmoins, certaines bases orthogonales ont été étudiées plus exhaustivement que d'autres, raison pour laquelle nous avons choisis dans ce présent ouvrage de présenter nos travaux à travers les polynômes orthogonaux discrets de Meixner, qui n'ont pas été abordés souvent par les chercheurs, et ce bien sûr en comparant les résultats obtenus avec d'autres bases orthogonales.

L'étude des moments au-delà des deux dimensions été très rare, seuls Sadjadi et Hall [Sadjadi, 1976] ont abordé les moments pour les volumes en 3D. C'est l'une des motivations pour ce travail que nous avons mené en collaboration avec différents collègues. L'imagerie 3D est devenue incontournable dans des domaines de pointes comme la médecine et l'astrophysique (ainsi que d'autres), mais le traitement des volumes 3D est resté assez couteux en temps et en mémoire. Mieux encore, extraire les descriptifs d'un volume 3D pour la reconstruction et la classification serait un apport technique très considérable.

Parmi les problématiques différentes que nous avons la possibilité d'étudier, nous avons choisis la plus primordiale, celle du temps de calcul. Les algorithmes proposés, pour extraire les moments orthogonaux discrets d'une image sont très couteux en temps, même ceux proposés jusque là pour accélérer les calculs, sont restés inefficaces, alors qu'ils avoisinent '1seconde' pour extraire les moments d'une image de taille 128x128, on peut dire que dans le domaine du « temps réel » que ces algorithmes sont obsolètes et ne sont d'aucune réel utilité dans la vrai vie ou quand il faut exécuter ces calculs sur une base des données de milliers d'images, encore pire quant il s'agît d'effectuer des calculs pour une image 3D, que peut prendre des dizaines de minutes , rien que pour une seule image de taille 64x64x64 et pour des ordres très petits.

Le souci d'accélérer les calculs, n'est pas le plus important, mais à notre avis le plus primordial, ainsi une méthode proposé peut prendre de la valeur, et peut être envisageable

pour de réelles applications. Ils vont de pair, la rapidité et la précision, et là nous voulons bien dire, la précision des calculs par les réductions des erreurs et de perte d'information, ainsi la reconstruction d'une image doit s'approcher au plus de l'image original, seulement là nous pourront considérer qu'un algorithme rapide de calculs des moments est efficace.

Les moments ont été principalement utilisés pour étudier, la reconstruction d'image, la reconnaissance des formes, la classification des images,…Dans ce travail nous présentons aussi une tentative de classification via les moments invariants de Meixner pour les images 3D, afin de donner plus de crédibilité à l'efficacité de nos algorithmes proposés.

Répartition de l'ouvrage :

Dans ce présent ouvrage, nous avons opté pour la répartition suivante :

- Chapitre 1 : Etat d'art, présentations des moments d'image et des polynômes orthogonaux, à travers une revue de travaux précédents.
- Chapitre 2 : Première approche pour l'accélération des calculs à travers l'introduction de la formule de récurrence de Clenshaw sur des images 2D
- Chapitre 3 : deuxième approche pour l'accélération des calculs à travers l'introduction de représentation des images 2D par des blocs.
- Chapitre 4 : troisième approche, accélérer les calculs des moments orthogonaux sur les images 3D, où nous introduisons pour la première fois une notion que nous avons nommé « la représentation des images 3D par les cuboïdes », qui sont des blocs 3D de voxels homogènes de même intensité. Puis nous avons étudié la classification des images 3D en exploitant l'approche proposée et à travers certaines caractéristiques invariantes (Translation, rotation, et mise à l'échelle).

Avant de conclure cet ouvrage par les perspectives que lancent notre travail pour de futurs recherches et applications.

CHAPITRE 1 : GENERALITE SUR LES POLYNOMES ET LES MOMENTS CONTINUES ET DISCRETS

Les polynômes orthogonaux jouent un rôle central dans la résolution numérique des équations aux dérivées partielles. Ils interviennent notamment dans le calcul des moments orthogonaux où ils sont utilisés dans plusieurs domaines de l'analyse d'image à savoir la reconstruction, la compression, la reconnaissance de forme et la classification des objets. Dans ce chapitre, nous allons rappeler quelques propriétés des polynômes orthogonaux discrets et continus. Nous présenterons ensuite les déférentes méthodes de calcul des valeurs des polynômes à savoir le calcul direct à partir des fonctions hyper géométriques et le calcul récursif utilisant les relations de récurrence sur l'ordre n et sur la variable x.

En général les moments d'ordre (p+q) d'une fonction *f(x,y)* sont définis par :

$$M_{p,q} = \iint_{Df} P_{pq}(x,y) f(x,y) dxdy \qquad \textbf{(I.1)}$$

Où $p,q=0,1,2,...,\infty$ et les polynômes $P_{pq}(x,y)$ sont des fonctions polynomiales de base définies sur $Df \subset \mathbb{R}\times\mathbb{R}$.

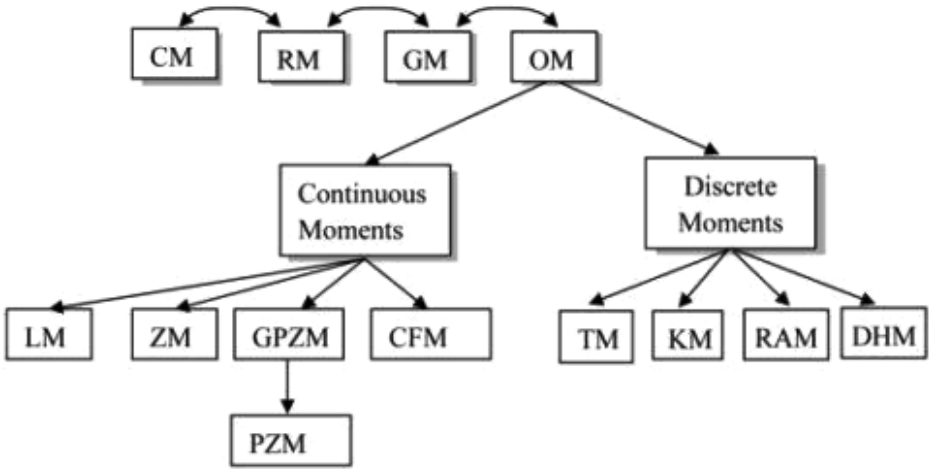
Selon la base des polynômes utilisée, nous définissons les différents types des moments.

Cette définition est extensible pour les volume en 3D *f(x,y,z)* , tel que SADJADI [SADJADI, 1980] l'a définit, le moment d'ordre *(p+q+r)* peut être représenté comme suit :

$$M_{p,q,r} = \int\iint_{Df} P_{pqr}(x,y,z) f(x,y,z) dxdydz \qquad \textbf{(I.2)}$$

Où $p,q,r = 0,1,2,...,\infty$ et les polynômes $P_{pqr}(x,y,z)$ sont des fonctions polynomiales de base définies sur $Df \subset \mathbb{R}\times\mathbb{R}\times\mathbb{R}$.

Les moments sont répartis en deux grandes familles, les moments non-orthogonaux et les moments orthogonaux, ainsi nous étudierons aussi les moments orthogonaux continues et discrets, la figure (I.1) montre l'arborescence des moments étudiés jusque là.



CM: Complex moments; RM: Rotational Moments; GM: Geometric Moments; OM: Orthogonal Moments; LM: Legendre Moments; ZM: Zernike Moments; PZM: Pseudo-Zernike Moments; GPZM: Generalized pseudo-Zernike Moments; CFM: Chebychev-Fourier Moments; TM: Tchebichef Moments; KM: Krawtchouk Moments; RAM: Racah Moments; DHM: Dual Hahn Moments

Figure (I.1): famille des moments

Nous estimons qu'il n'y a pas besoins d'être exhaustif, alors nous présentons dans ce chapitre que nous considérons comme état d'art de cet ouvrage, seulement les moments orthogonaux. Plusieurs ouvrages discutent en détails les moments non-orthogonaux (géométriques, complexes, et rotationnels), ils seront cités plus tard dans cet ouvrage pour éclairer certains raisonnement et calculs mathématiques, notamment lorsque nous aborderont la classification des images 3D.

1. Généralités sur les Polynômes orthogonaux à une seule variable

Dans cette section, nous allons définir la notion d'orthogonalité des polynômes au sens de la mesure de Lebesgue-Stieltjes dans un intervalle.

1.1. Notion d'orthogonalité

Une séquence des polynômes $\{p_n\}_{n\in N}$, où p_n est un polynôme de degré n en x, est dite orthogonale par rapport à la mesure de Lebesgue-Stieltjes $d\mu(x)$ sur l'intervalle $[a,b]$ si :

$$\int_a^b p_m(x)p_n(x)d\mu(x)=0, \quad m\neq n \tag{I.3}$$

Si la mesure μ est bornée et absolument continue sur $[a,b]$ tel que $d\mu(x)=w(x)dx$, on dit que les polynômes orthogonaux $\{p_n\}_{n\in N}$ sont d'une variable continue et la relation d'orthogonalité définie en (I.3) devient [Suslov, 1989]:

$$\int_a^b p_m(x)p_n(x)w(x)dx=0, \quad m\neq n \tag{I.4}$$

Si la mesure μ est une fonction en escalier à des sauts w_j au point $x=x_j$, $j=0,1,2,...$., nous dirons que les polynômes orthogonaux $\{p_n\}_{n\in N}$ sont d'une variable discrète et la relation d'orthogonalité définie en (I.3) prend la forme suivante [Suslov, 1989]:

$$\sum_{j=0}^{\infty} p_m(x_j)p_n(x_j)w_j(x_j)=0, \quad m\neq n \tag{I.5}$$

1.2. Procédé d'orthogonalisation

En appliquant le procédé d'orthogonalisation de Gram-Schmidt [Szego, 1959], on obtient un ensemble fini des polynômes orthogonaux $p_0,p_1,...,p_{n-1}$ et chaque polynôme de degré n peut s'écrire comme une combinaison linéaire des polynômes $p_0,p_1,...,p_{n-1}$. Donc, pour chaque $m\in N$, le polynôme p_m est orthogonal à tous les polynômes de degré $k\prec m$; en particulier nous écrivons :

$$\int p_n(x)x^k d\mu(x) = 0, \ \ pour\ tout \quad k = 0,1,...,n-1 \tag{I.6}$$

En effet, cette condition détermine le polynôme p_n à une constante près. Si nous considérons que les coefficients du polynôme p_n comme des variables, l'équation (I.6) détermine un système linéaire avec n équations et $n+1$variables. Si nous fixons la constante de normalisation, nous ajoutons une équation de plus à ce système et donc, lors de la résolution de ce système, on obtient une solution unique en termes de la mesure μ .

1.3. Formules de récurrence

La propriété d'orthogonalité implique une série de propriétés importantes pour les polynômes telles que les relations de récurrence par rapport à l'ordre n [Sezgo, 1959]. Il existe deux types de relations de récurrences suivant la nature de la base des polynômes : orthogonale ou orthonormée.

1.3.1. Base orthogonale

Considérons un système de trois polynômes orthogonaux consécutifs $p_{m-1}(x),\ p_m(x)\ et\ p_{m+1}(x)$, la relation de récurrence est définie dans ce cas par [Sezgo, 1959] :

$$P_{m+1}(x) = (A_m x + B_m)P_m(x) - C_m P_{m-1}(x) \quad ; \ m = 1,2,3,... \tag{I.7}$$

où A_m, B_m et C_m sont des constantes définies par :

$$A_m = \frac{a_{m+1}}{a_m}, \ C_m = \frac{a_{m+1}a_{m-1}h_m}{a_n^2 h_{m-1}}, \qquad B_m = -\frac{a_{m+1}}{a_n h_m}\int_a^b x p_n^2(x) d\mu(x),$$

$$h_n = \int_a^b p_n^2(x) d\mu(x)$$

et a_m est le principal coefficient du polynôme p_m .

Dans le but de simplifier encore cette formule nous allons, en plus de la condition de normalisation, exiger que chaque polynôme doit être unitaire ou le terme constant du

polynôme égal à un. En effet, pour le cas particulier où les polynômes orthogonaux sont unitaires, la relation de récurrence des trois termes définie par l'équation (I.7) devient :

$$P_{m+1}(x) = (x + B_m)P_m(x) - C_m P_{m-1}(x) \quad ; \; m = 1,2,3,... \tag{I.8}$$

où A_m, B_m et C_m sont des constantes définies par :

$$A_m = 1 \qquad ; C_m = \frac{h_m}{h_{m-1}}, \qquad B_m = -\frac{1}{h_m}\int_a^b x p_n^2(x) d\mu(x)$$

1.3.2. Base orthonormée

Pour avoir la relation de récurrence à trois termes pour une base des polynômes orthonormés $\{p_m\}_{m\in N}$, il suffit de diviser les polynômes $\{p_m\}_{m\in N}$ par leurs normes carrées. On obtient ainsi une famille des polynômes orthonormés $\tilde{P}_n = \frac{P_n}{\|P_n\|}$, où $\|P_n\| = \int_a^b P^2(x) d\mu(x)$ est la norme associée au produit scalaire. Cette famille vérifie alors la relation de récurrence suivante:

$$x p_m(x) = A_m p_{m+1}(x) + B_m p_m(x) - A_{m-1} p_{m-1}(x) \qquad , \quad m = 2,3,.... \tag{I.9}$$

où $A_n = \frac{a_n}{a_{n+1}}$, $B_n = \int_a^b x p_n^2(x) d\mu(x)$

1.4. Formule Christoffel-Darboux

Le calcul explicite des polynômes orthogonaux ainsi que l'étude de la distribution des zéros de ces polynômes passent par la formule de Christoffel-Darboux [Nikiforov et al., 1991]. Cette dernière s'obtient par combinaison des différentes relations de récurrence citées avant et définie par :

$$\sum_{k=0}^{m} \frac{p_k(x)p_k(y)}{\|p_k\|^2} = \frac{p_{m+1}(x)p_m(y) - p_m(x)p_{m+1}(y)}{A_m \|p_m\|^2 (x-y)}, \quad m \in N \quad et \quad (x,y) \in [a,b]^2 \tag{I.10}$$

1.5. Types de polynômes orthogonaux

D'après les relations d'orthogonalités des polynômes, le choix de la fonction de poids $w(x)$ permet d'obtenir plusieurs ensembles des polynômes orthogonaux. Ils satisfont, selon la nature de la variable (continue ou discrète), les équations différentielles du second ordre ou les équations de différence à coefficients polynomiales [Nikiforov et al., 1991].

1.5.1. Polynômes orthogonaux d'une variable continue

Les polynômes orthogonaux d'une variable continue satisfont l'équation différentielle du second ordre suivante:

$$\sigma(x)\frac{d^2}{dx^2}y(x)+\tau(x)\frac{d}{dx}y(x)+\lambda_n y(x)=0 \qquad \textbf{(I.11)}$$

avec σ et τ sont des polynômes de 2iéme degré et du premier degré. λ_n est une constante qui dépend des fonctions σ et τ.

Les polynômes solutions de cette équation sont les polynômes de Jacobi, Laguerre, Hermite et Bessel.

1.5.2. Polynômes orthogonaux d'une variable discrète

Les polynômes orthogonaux d'une variable discrète sur un intervalle linéaire de la forme : $x_j = j \quad , \quad j = 0,1,...,$ sont solutions de l'équation de différence du second ordre suivante :

$$\sigma(x)\Delta\nabla y(x)+\tau(x)\nabla y(x)+\lambda_n y(x)=0 \qquad \textbf{(I.12)}$$

où $\sigma(x)$ et $\tau(x)$ sont des polynômes du deuxième et premier degré, respectivement. λ_n est une constante appropriée. Les opérateurs Δ et ∇ indiquent respectivement, l'opérateur de différence en avance et l'opérateur de différence en arrière. Ces derniers sont définis comme suit :

$$\Delta y(x)=y(x+1)-y(x) \quad ; \quad \nabla y(x)=y(x)-y(x-1)$$

Les polynômes solutions de l'équation (I.12) sont principalement les polynômes de Tchebichef, Hahn, Krawtchouk, Charlier et Meixner.

Le schéma Askey suivant représente les polynômes orthogonaux classiques et leurs relations de la limite [Koekoek & Swarttouw, 1998].

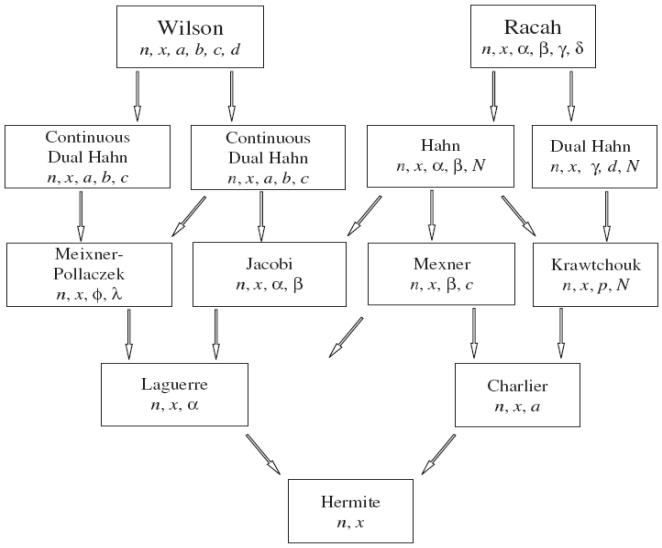


Figure (I.2): Le schéma Askey pour les polynômes hypergéométriques orthogonaux, avec les relations de la limite indiquées entre les polynômes.

Dans le paragraphe précèdent les polynômes orthogonaux ont été définis comme solutions des équations différentielle ou de différences. En littérature ces polynômes peuvent être définis à partir des fonctions hypergéométriques. Ces derniers seront présentés dans le paragraphe suivant.

1.6. Série hypergéométrique de base

La série hypergéométrique généralisée ${}_rF_s$ est définie par :

$$ {}_rF_s\left(\begin{matrix}a_1,\ldots,a_r\\ b_1,\ldots,b_s\end{matrix},x\right)=\sum_{k=0}^{\infty}\frac{(a_1)_k\ldots(a_r)_k\,x^k}{(b_1)_k\ldots(b_s)_k\,k!} \qquad \textbf{(I.13)} $$

où $(a)_k$ le symbole de Pochhammer défini par :

$$(a)_k = a(a+1)...(a+k-1) = \frac{\Gamma(a+k)}{\Gamma(a)} \qquad \textbf{(I.14)}$$

et Γ est la fonction gamma définie par : $\Gamma(x) = \int_0^\infty e^{-t} t^{x-1} dt$

Par une simple intégration par partie de la fonction gamma, on trouve la relation de récurrence suivante : $\Gamma(x+1) = x\Gamma(x)$ pour tout $x \geq 0$ et $\Gamma(n+1) = n!$ pour tout entier n .

La notation hypergéométrique permet de regrouper un nombre infini des fonctions spéciales. En effet, en substituant les arguments de ${}_rF_s\left(\begin{matrix} a_1,...,a_r \\ b_1,...,b_s \end{matrix}, x\right)$ par des valeurs particulières, on reconnaît des fonctions qui sont largement appliquées en physique.

A titre d'exemple, on donne des exemples des fonctions hypergéométriques :

$${}_0F_0\left(\begin{matrix} - \\ - \end{matrix}, x\right) = e^x$$

$${}_1F_0\left(\begin{matrix} a \\ - \end{matrix}, x\right) = (1-x)^{-a}$$

$${}_0F_1\left(\begin{matrix} - \\ 1/2 \end{matrix}, \frac{-x^2}{4}\right) = \cos x$$

$${}_2F_1\left(\begin{matrix} 1,1 \\ 2 \end{matrix}, -x\right) = \frac{1}{x}\log(x+1)$$

1.6.1. Propriétés élémentaires des séries hypergéométriques

La série hypergéométrique généralisée ${}_rF_s\left(\begin{matrix} a_1,...,a_r \\ b_1,...,b_s \end{matrix}, x\right)$ est bien définie si les paramètres b_i du dénominateur sont différents de zéro.

Si l'un des paramètres de numérateur a_i de la série hypergéométrique ${}_rF_s\left(\begin{matrix}a_1,...,a_r\\ b_1,...,b_s\end{matrix},x\right)$ est égal à $-n$ où n est un entier positif ou nul, cette série est un polynôme de variable x.

Sinon, c'est-à-dire $\forall i \ , a_i \neq 0$, la série hypergéométrique est convergente et son rayon de convergence ρ est défini par [Koekoek & Swarttouw, 1998] :

$$\rho = \begin{cases} \infty & if \quad r < s+1 \\ 1 & if \quad r = s+1 \\ 0 & if \quad r > s+1 \end{cases} \qquad \textbf{(I.15)}$$

1.6.2. Opérations sur les séries hypergéométriques

- La dérivée de la série hypergéométrique ${}_rF_s$ est définie par :

$$\frac{d\,{}_rF_s}{dx} = \frac{a_1.a_2...a_r}{b_1.b_1...b_s}\,{}_rF_s\left(\begin{matrix}a_1+1,a_2+1,...,a_r+1\\ b_1+1,b_1+1,...,b_s+1\end{matrix}\middle| x\right) \qquad \textbf{(I.16)}$$

- L'intégrale de la série hypergéométrique ${}_rF_s$ est définie par

$$\int_0^x {}_rF_s\left(\begin{matrix}a_1,...,a_r\\ b_1,...,b_s\end{matrix}\middle| x\right)dx = \frac{(b_1-1)(b_2-1)...(b_s-1)}{(a_1-1)(a_2-1)...(a_r-1)}\left[{}_rF_s\left(\begin{matrix}a_1-1,a_2-1,...,a_r-1\\ b_1-1,b_1-1,...,b_s-1\end{matrix}\middle| x\right)-1\right] \qquad \textbf{(I.17)}$$

- La transformée de Laplace de la série hypergéométrique ${}_rF_s$ est définie par :

$$\int_0^x e^{-st}t^{v-1}\,{}_rF_s\left(\begin{matrix}a_1,...,a_r\\ b_1,...,b_s\end{matrix}\middle| \lambda t\right)dt = s^{-v}\Gamma(v)\,{}_{r+1}F_s\left(\begin{matrix}a_1,a_2,...,a_r,v\\ b_1,b_1,...,b_s\end{matrix}\middle| \frac{\lambda}{s}\right)$$

si $r < s, \quad v > 0$

si $r = s, \quad v > 0 \ et \ s < \lambda$ **(I.18)**

- La série hypergéométrique ${}_rF_s$ satisfait l'équation différentielle suivante :

$$\left[x\frac{d}{dx}\prod_{j=1}^{s}(x\frac{d}{dx}+b_j+1) - x\prod_{j=1}^{r}(x\frac{d}{dx}+a_j)\right]{}_rF_s\left(\begin{matrix}a_1,...,a_r\\ b_1,...,b_s\end{matrix},x\right) = 0 \qquad \textbf{(I.19)}$$

2. Polynômes et moments continues

2.1. Polynômes orthogonaux d'une variable continues

Le traitement d'un grand nombre de problèmes de la physique théorique et des mathématiques passe par la résolution de l'équation différentielle suivante :

$$\sigma(x)\frac{d^2}{dx^2}y(x)+\tau(x)\frac{d}{dx}y(x)+\lambda_n y(x)=0 \qquad \textbf{(I.20)}$$

Où σ et τ sont des polynômes de 2iéme degré et du premier degré, respectivement.

λ_n est une constante définie par : $\lambda_n=-n\tau'(x)-\frac{1}{2}n(n-1)\sigma''(x)$

Cette équation admet comme solutions particulières des fonctions spéciales appartenant aux classes des fonctions hypergéométriques et des polynômes orthogonaux d'une variable continue.

Les polynômes solutions de cette équation sont donnés explicitement par la formule de Rodriguez [Nikiforov et al., 1991]:

$$y_n(x)=\frac{B_n}{w(x)}\left[\sigma^n(x)w(x)\right]^{(n)} \qquad \textbf{(I.21)}$$

où B_n est une constante de normalisation.

La fonction de poids $w(x)$ satisfait l'équation différentielle du premier ordre (appelée équation de Pearson) suivante :

$$\frac{d}{dx}(\sigma(x)w(x))=\tau(x)w(x) \qquad \textbf{(I.22)}$$

En résolvant l'équation de Pearson (I.22), nous obtenons trois formes possibles de la fonction de poids $w(x)$ selon le degré du polynôme $\sigma(x)$:

$$w(x)=\begin{cases}(b-x)^{\alpha}(x-a)^{\beta} & pour \quad \sigma(x)=(b-x)(x-a)\\ (x-a)^{\alpha}e^{\beta x} & pour \quad \sigma(x)=x-a\\ e^{\alpha x^2+\beta x} & pour \quad \sigma(x)=1\end{cases} \tag{I.23}$$

où a, b, α et β sont des constantes (complexes dans le cas général).

Par un changement linéaire de la variable indépendante, les expressions des fonctions $\sigma(x)$ et $w(x)$ peuvent être réduites pour obtenir les formes canoniques suivantes.

$$w(x)=\begin{cases}(1-x)^{\alpha}(1+x)^{\beta} & pour \quad \sigma(x)=1-x^2\\ x^{\alpha}e^{-x} & pour \quad \sigma(x)=x\\ e^{-x^2} & pour \quad \sigma(x)=1\end{cases} \tag{I.24}$$

Sous ces transformations les équations (I.20) et (I.22) deviennent des équations de la même forme, et les polynômes correspondants $y_n(x)$ restent de types hypergéométriques sur la nouvelle variable et sont définis par la formule Rodriguez (I.21).

Selon la forme des fonctions $\sigma(x)$, $\tau(x)$ et $w(x)$, on obtient les polynômes orthogonaux d'une variable continue, que nous présentons en sous-sections.

2.1.1. Les polynômes de Jacobi

Les polynômes orthogonaux d'une variable continue de Jacobi satisfont l'équation différentielle (I.26) sous les conditions suivantes [Nikiforov et al., 1991] :

$$\begin{aligned}&\sigma(x)=1-x^2,\ \ w(x)=(1-x)^{\alpha}(1+x)^{\beta} \text{ et } \ \tau(x)=-(\alpha+\beta+2)x+\beta-\alpha\\ &pour \quad \alpha \succ -1 \quad et \quad \beta \succ -1\end{aligned} \tag{I.25}$$

Les polynômes orthogonaux de Jacobi $J_n^{(\alpha,\beta)}(x)$ sont définis à partir de la formule de Rodriguez sous la condition de normalisation $B_n=\dfrac{(-1)^n}{2^n n!}$ par la relation suivante :

$$J_n^{(\alpha,\beta)}(x)=\frac{(-1)^n}{2^n n!}(1-x)^{-\alpha}(1+x)^{-\beta}\frac{d^n}{dx^n}\left[(1-x)^{n+\alpha}(1+x)^{n+\beta}\right], \quad x\in\left]-1,1\right[\tag{I.26}$$

En effectuant la dérivée nième du produit, nous obtenons une expression explicite des polynômes de Jacobi $J_n^{(\alpha,\beta)}(x)$.

$$J_n^{(\alpha,\beta)}(x)=\frac{\Gamma(\alpha+n+1)}{n!\Gamma(\alpha+\beta+n+1)}\sum_{m=0}^{n}\binom{n}{m}\frac{\Gamma(\alpha+\beta+n+m+1)}{\Gamma(\alpha+m+1)}\left(\frac{x-1}{2}\right)^m \quad \textbf{(I.27)}$$

Les polynômes de Jacobi d'ordre n, sont définis explicitement comme une somme hypergéométrique généralisée par la relation suivante [Koekoek & Swarttouw, 1998] :

$$J_n^{(\alpha,\beta)}(x)=\frac{(\alpha+1)_n}{n!}\,{}_2F_1\left(\begin{matrix}-n,n+\alpha+\beta+1\\ \alpha+1\end{matrix}\middle|\frac{1-x}{2}\right) \quad \textbf{(I.28)}$$

Les polynômes de Jacobi sont orthogonaux sur l'intervalle $]-1,1[$ relativement à la fonction de poids $w(x)=(1-x)^{\alpha}(1+x)^{\beta}$ et vérifiant la relation d'orthogonalité suivante :

$$\int_{-1}^{1}J_n^{(\alpha,\beta)}(x)J_m^{(\alpha,\beta)}(x)w(x)dx=d_n^2\delta_{n,m} \quad \textbf{(I.29)}$$

où d_n^2 est la fonction norme carrée définie comme suit :

$$d_n^2=\frac{2^{\alpha+\beta+1}}{2n+\alpha+\beta+1}\frac{\Gamma(n+\alpha+1)\Gamma(n+\beta+1)}{n!\Gamma(n+\alpha+\beta+1)} \quad \textbf{(I.30)}$$

et $\delta_{n,m}$ est le symbole de Kronecker défini par :

$$\delta_{n,m}=\begin{cases}1 & \text{si} \quad n=m\\ 0 & \text{si} \quad n\neq m\end{cases} \quad \textbf{(I.31)}$$

Pour éviter l'instabilité numérique lors du calcul des valeurs des polynômes, on utilise les polynômes de Jacobi normalisés par la norme carrée et la fonction de poids. L'ensemble des polynômes normalisés de Jacobi est défini par la relation suivante :

$$\tilde{J}_n^{(\alpha,\beta)}(x)=J_n^{(\alpha,\beta)}(x)\sqrt{\frac{w(x)}{d_n^2}} \quad \textbf{(I.32)}$$

Le calcul des polynômes de Jacobi, basé sur les fonctions hypergéométriques (I.28) et les fonctions gamma, est très coûteux en temps de calcul. Pour remédier à ce problème nous utilisons la relation de récurrence des polynômes normalisés de Jacobi donnée par [Koekoek & Swarttouw, 1998]:

$$J_{n+1}^{(\alpha,\beta)}(x) = (x - \beta_n) J_n^{(\alpha,\beta)}(x) - \gamma_n J_{n-1}^{(\alpha,\beta)}(x) \tag{I.33}$$

avec

$$\beta_n = \frac{\beta^2 - \alpha^2}{(2n+\alpha+\beta)(2n+\alpha+\beta+2)}, \tag{I.34}$$

$$\gamma_n = \frac{4n(n+\alpha)(n+\beta)(n+\alpha+\beta)}{(2n+\alpha+\beta-1)(2n+\alpha+\beta)^2(2n+\alpha+\beta+1)} \tag{I.35}$$

La figure (I.3) représente les courbes des six premiers ordres des polynômes de Jacobi avec différentes valeurs des paramètres α *et* β.

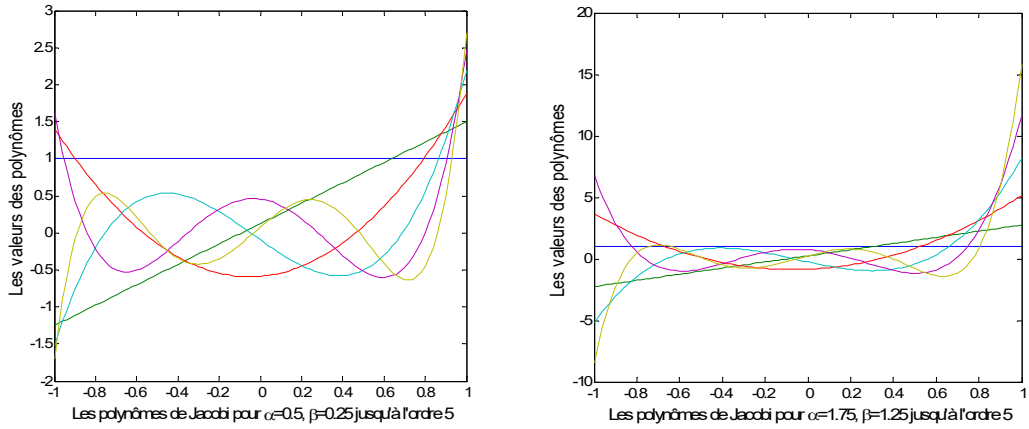


Figure (I.3): Courbes des polynômes orthogonaux de Jacobi pour $\alpha = 0.5$ *et* $\beta = 0.25$ et $\alpha = 1.75$ *et* $\beta = 1.25$ pour les six premiers ordres.

Les polynômes de Jacobi recouvrent plusieurs cas particuliers intéressants des polynômes : les polynômes de Legendre, pour $\alpha = \beta = 0$, les polynômes de Chebyshev de première espèce pour ($\alpha = \beta = -1/2$), de deuxième espèce pour ($\alpha = \beta = 1/2$) et les

polynômes de Gegenbauer pour ($\alpha = \beta = \lambda - 1/2$), avec λ est un paramètre propre au polynôme de Gegenbauer.

2.1.2. Les polynômes de Legendre

Les polynômes de Legendre sont définis à partir de la formule de Rodriguez (I.26) pour $\alpha = \beta = 0$ par la relation suivante [Nikiforov et al., 1991]:

$$P_n(x) = J_n^{(0,0)}(x) = \frac{(-1)^n}{2^n n!}\frac{d^n}{dx^n}\left[\left(1-x^2\right)^n\right], \quad x \in]-1,1[\tag{I.36}$$

En effectuant la dérivée nième du produit, nous obtenons la forme hypergéométrique de polynômes de Legendre $P_n(x)$:

$$P_n(x) = \frac{(1)_n}{n!}\,{}_2F_1\left(\begin{matrix}-n, n+\alpha+\beta+1 \\ \alpha+1\end{matrix}\middle| \frac{1-x}{2}\right) \tag{I.37}$$

Les polynômes orthogonaux de Legendre satisfont la relation de récurrence à trois termes suivante :

$$\begin{aligned}&(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x)\\ &avec \quad P_0(x) = 1 \qquad et \qquad P_1(x) = x\end{aligned} \tag{I.38}$$

Les polynômes de Legendre satisfont la relation d'orthogonalité relativement à la fonction de poids $w(x) = 1$ sur l'intervalle $]-1,1[$:

$$\int_{-1}^{1} P_n(x)P_m(x)dx = \frac{2}{2n+1}\delta_{n,m} \tag{I.39}$$

La figure (I.4) représente les courbes des six premiers ordres des polynômes orthonormés de Legendre.

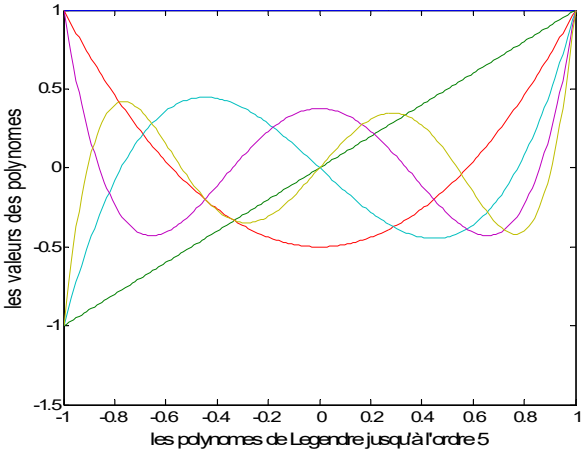


Figure (I.4) : Courbes des polynômes orthonormés de Legendre pour les six premiers ordres.

2.1.3. Les polynômes de Tschebyshev

Les polynômes de Chebyshev de première espèce $T_n(x)$ sont définis à partir de la formule de Rodriguez pour ($\alpha = \beta = -1/2$) par la relation suivante [Koekoek & Swarttouw, 1998] :

$$T_n(x) = \frac{n!}{(1/2)_n} P_n^{(-1/2,-1/2)}(x) = \frac{(-1)^n}{2^n (1/2)_n} (1-x^2)^{1/2} \frac{d^n}{dx^n}\left[(1-x^2)^{n-1/2}\right] \qquad \textbf{(I.40)}$$

Ceux de deuxième espèce $U_n(x)$ sont définis pour ($\alpha = \beta = 1/2$) par la relation suivante [Koekoek & Swarttouw, 1998] :

$$U_n(x) = \frac{(n+1)!}{(3/2)_n} P_n^{(1/2,1/2)}(x) = \frac{(-1)^n (n+1)}{2^n (3/2)_n} (1-x^2)^{1/2} \frac{d^n}{dx^n}\left[(1-x^2)^{n-1/2}\right] \qquad \textbf{(I.41)}$$

Les polynômes de Chebyshev de première espèce $T_n(x)$ et de deuxième espèce $U_n(x)$ sont définis en termes des fonctions hypergéométriques par les relations suivantes :

$$T_n(x)=\frac{P_n^{(-1/2,-1/2)}(x)}{P_n^{(-1/2,-1/2)}(1)}={}_2F_1\left(\begin{matrix}-n,n\\ \frac{1}{2}\end{matrix}\middle|\frac{1-x}{2}\right)$$
$$U_n(x)=(n+1)\frac{P_n^{(1/2,1/2)}(x)}{P_n^{(1/2,1/2)}(1)}=(n+1)\,{}_2F_1\left(\begin{matrix}-n,n+2\\ \frac{3}{2}\end{matrix}\middle|\frac{1-x}{2}\right) \qquad \textbf{(I.42)}$$

Les polynômes orthogonaux de Chebyshev de première espèce $T_n(x)$ et de deuxième espèce $U_n(x)$ satisfont les relations de récurrences à trois termes suivantes :

$$\begin{aligned}&2xT_n(x)=T_{n+1}(x)+T_{n-1}(x),\quad T_0(x)=1\ \ et\ \ T_1(x)=x\\&2xU_n(x)=U_{n+1}(x)+U_{n-1}(x),\quad U_n(x)=1\ \ et\ \ U_n(x)=2x\end{aligned} \qquad \textbf{(I.43)}$$

Les polynômes de Chebyshev de première espèce $T_n(x)$ et de deuxième espèce $U_n(x)$ sont des polynômes orthogonaux relativement à la fonction de poids $w(x)=1$ sur l'intervalle $]-1,1[$ et vérifient la relation d'orthogonalité suivante :

$$\int_{-1}^{1}\left(1-x^2\right)^{-\frac{1}{2}}T_n(x)T_m(x)dx=\begin{cases}\frac{\pi}{2}\delta_{n,m}\\ \pi\delta_{n,m}\end{cases}$$
$$\int_{-1}^{1}\left(1-x^2\right)^{-\frac{1}{2}}U_n(x)U_m(x)dx=\frac{\pi}{2}\delta_{n,m} \qquad \textbf{(I.44)}$$

La figure (I.5) représente les courbes des six premiers ordres des polynômes orthogonaux de Chebyshev de première espèce et de deuxième espèce.

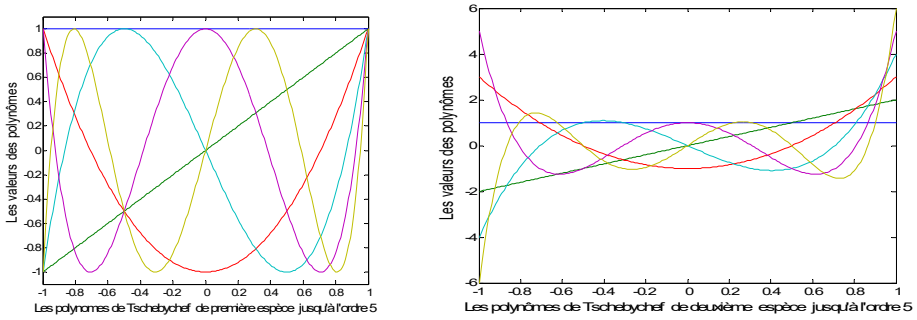


Figure (I.5) : Courbes des polynômes orthonormés de Chebyshev de première espèce et de deuxième espèce pour les six premiers ordres.

2.1.4. Les polynômes de Gegenbauer ou ultrasphériques

Les polynômes orthogonaux de Gegenbauer sont définis à partir de la formule de Rodriguez (I.26) pour $\alpha = \beta = \lambda - 1/2$ par la relation suivante [Nikiforov et al., 1991]:

$$C_n^{\lambda}(x) = \frac{(2\lambda)_n}{(\lambda+1/2)_n} \times \frac{(-1)^n}{2^n n!} \left(1-x^2\right)^{\frac{1}{2}-\lambda} \frac{d^n}{dx^n}\left[\left(1-x^2\right)^{\lambda+n-1/2}\right] \tag{I.45}$$

Les polynômes de Gegenbauer sont définis en termes les fonctions hypergéométriques par la relation suivante :

$$C_n^{(\lambda)}(x) = \frac{(2\lambda)_n}{n!}\, {}_2F_1\left(\begin{matrix} -n, n+2\lambda \\ \lambda+\frac{1}{2} \end{matrix}\middle| \frac{1-x}{2}\right), \quad \lambda \neq 0 \tag{I.46}$$

Les polynômes orthogonaux de Gegenbauer satisfont la relation de récurrence à trois termes suivante :

$$\begin{aligned} 2(n+\lambda)xC_n^{(\lambda)}(x) &= (n+1)C_{n+1}^{(\lambda)}(x) + (n+2\lambda-1)C_{n-1}^{(\lambda)}(x) \\ C_0^{(\lambda)}(x) = 1 \;\; et \;\; C_1^{(\lambda)}(x) &= 2\lambda x \;\; \text{si} \;\; \lambda \neq 0 \,,\; C_1^{(0)}(x) = 2x \end{aligned} \tag{I.47}$$

Les polynômes de Gegenbauer sont orthogonaux relativement à la fonction de poids $w(x) = \left(1-x^2\right)^{\lambda-\frac{1}{2}}$ sur l'intervalle $]-1,1[$ et vérifient la relation d'orthogonalité suivante :

$$\int_{-1}^{1} \left(1-x^2\right)^{\lambda-\frac{1}{2}} C_n^{(\lambda)}(x) C_m^{(\lambda)}(x) dx = \frac{\pi 2^{1-2\lambda}}{n!(n+\lambda)!} \frac{\Gamma(n+2\lambda)}{\Gamma(\lambda)^2} \delta_{n,m} \tag{I.48}$$

La figure (I.6) représente les courbes des six premiers ordres des polynômes orthogonaux de Gegenbauer avec différentes valeurs de paramètre λ.

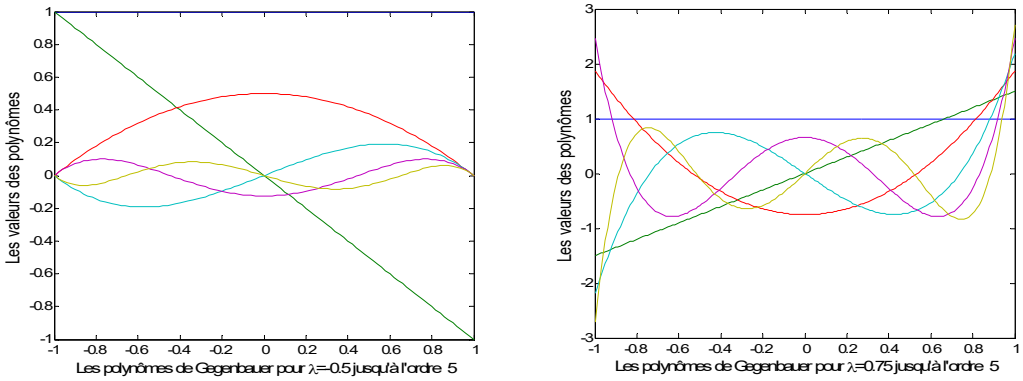


Figure (I.6): Courbes des polynômes orthogonaux de Gegenbauer ($\lambda = -0.5$) et ($\lambda = 0.75$) pour les six premiers ordres.

2.1.5. Les polynômes de Laguerre

Les polynômes de Laguerre satisfont l'équation différentielle de type hypergéométrique (I.26) sous les conditions :

$$\sigma(x) = x, \quad w(x) = x^{\alpha} e^{-x} \ , \quad \tau(x) = \alpha + 1 - x \quad \text{et } \lambda_n = n \quad pour \quad \alpha \succ -1 \tag{I.49}$$

Les polynômes orthogonaux de Laguerre $L_n^{(\alpha)}(x)$ sont définis à partir de la formule de Rodriguez sous la condition de normalisation $B_n = \frac{1}{n!}$ comme suit :

$$L_n^{(\alpha)}(x) = \frac{e^x}{n! x^{\alpha}} \frac{d^n}{dx^n}(x^{n+\alpha} e^{-x}), \qquad x \in [0, +\infty[\tag{I.50}$$

En effectuant la dérivée nième du produit, nous obtenons une formule explicite des polynômes de Laguerre $L_n^{(\alpha)}(x)$:

$$L_n^{(\alpha)}(x) = \sum_{k=0}^{n} \frac{(-1)^k \Gamma(\alpha + n + 1)}{(n-k)! k! \Gamma(\alpha + k + 1)} x^k \tag{I.51}$$

Les polynômes de Laguerre d'ordre n, sont définis explicitement comme une somme hypergéométrique généralisée par [Abramowitz & Stegun, 1970]:

$$L_n^{(\alpha)}(x) = \frac{(\alpha+1)_n}{n!} {}_1F_1(-n; \alpha + 1; x) \tag{I.52}$$

Les polynômes de Laguerre sont orthogonaux sur l'intervalle $[0,+\infty[$ relativement à la fonction de poids $w(x)=x^{\alpha}\mathrm{e}^{-x}$ et vérifient la relation d'orthogonalité suivante :

$$\int_{-1}^{1} L_n^{(\alpha)}(x)L_n^{(\alpha)}(x)w(x)dx = d_n^2 = \frac{\Gamma(n+\alpha+1)}{n!}\delta_{n,m} \qquad \textbf{(I.53)}$$

Les polynômes normalisés de Laguerre sont définis par :

$$\tilde{L}_n^{(\alpha)}(x) = \sqrt{\frac{n!}{\Gamma(\alpha+n+1)}}x^{\frac{\alpha}{2}}e^{-\frac{x}{2}}L_n^{(\alpha)}(x) \qquad \textbf{(I.54)}$$

Afin d'accélérer le temps de calcul des polynômes de Laguerre, nous utilisons la relation de récurrence des polynômes normalisés de Laguerre définis par :

$$\begin{aligned}&(n+1)L_{n+1}^{\alpha}(x) = (2n+\alpha+1-x)L_n^{\alpha}(x) - (n+\alpha)L_{n-1}^{\alpha}(x)\\ &avec \qquad L_0^{\alpha}(x)=1 \qquad \text{et} \quad L_1^{\alpha}(x)=1+\alpha-x\end{aligned} \qquad \textbf{(I.55)}$$

La figure (I.7) représente les courbes des six premiers ordres des polynômes orthonormés de Laguerre avec différentes valeurs de paramètre α .

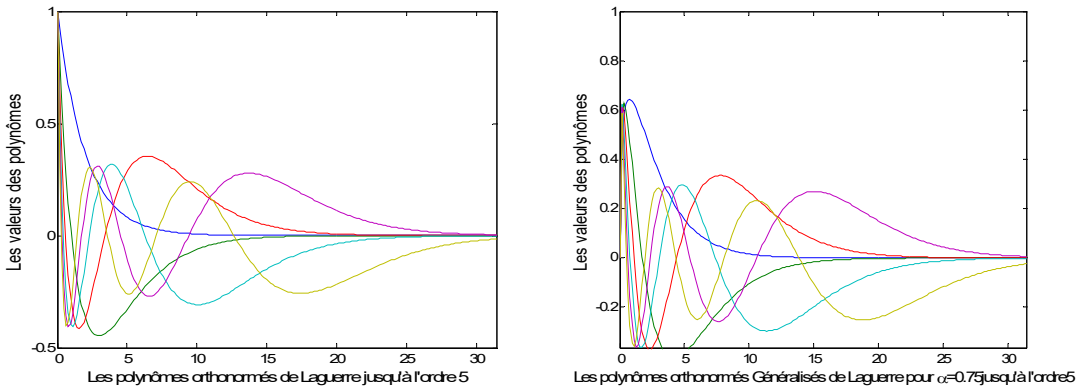


Figure (I.7): Courbes des polynômes orthonormés de Laguerre pour ($\alpha=0$) et ($\alpha=0.75$) pour les six premiers ordres.

2.1.6. Les polynômes d'Hermite

Les polynômes d'Hermite satisfont l'équation différentielle (I.26) sous les conditions suivantes [Koekoek & Swarttouw, 1998] :

$$\sigma(x)=1,\ \ w(x)=\mathrm{e}^{-x^2}\ ,\ \ \tau(x)=-2x\quad \text{et } \lambda_n=2n \tag{I.56}$$

Les polynômes orthogonaux d'Hermite $H_n(x)$ sont définis à partir de la formule de Rodriguez suivante :

$$H_n(x)=(-1)^n e^{x^2}\frac{d^n}{dx^n}\left(e^{-x^2}\right),\qquad x\in\left]-\infty,+\infty\right[\tag{I.57}$$

En effectuant la dérivée nième de la fonction e^{-x^2}, nous obtenons une formule explicite des polynômes d'Hermite $H_n(x)$:

$$H_n(x)=\sum_{k=0}^{n/2}\frac{(-1)^k n!}{(n-2k)!k!}(2x)^{n-2k} \tag{I.58}$$

Les polynômes d'Hermite d'ordre n, sont définis comme somme hypergéométrique généralisée par :

$$H_n(x)=x^n\ {}_2F_0\left(\begin{matrix}-\frac{n}{2},-\frac{n-1}{2}\\ -\end{matrix}\middle|-\frac{1}{x^2}\right) \tag{I.59}$$

Les polynômes d'Hermite sont orthogonaux sur l'intervalle $\left]-\infty,+\infty\right[$ relativement à la fonction de poids $w(x)=\mathrm{e}^{-x^2}$ et vérifient la relation d'orthogonalité suivante.

$$\int_{-1}^{1}H_n(x)H_m(x)\mathrm{e}^{-x^2}dx=\sqrt{\pi}\,2^n n!\delta_{n,m} \tag{I.60}$$

Les polynômes normalisés d'Hermite sont définis par la relation suivante :

$$\tilde{H}_n(x)=\sqrt{\frac{\mathrm{e}^{-x^2}}{\sqrt{\pi}\,2^n n!}}H_n(x) \tag{I.61}$$

La relation de récurrence des polynômes normalisés d'Hermite est définie par :

$$H_{n+1}(x) = 2xH_n(x) - 2nH_{n-1}(x), \qquad n \geq 1$$
$$avec \quad H_0(x) = 1 \quad et \quad H_1(x) = 2x \tag{I.62}$$

La figure (I.8) représente les courbes des six premiers ordres des polynômes orthonormés d'Hermite.

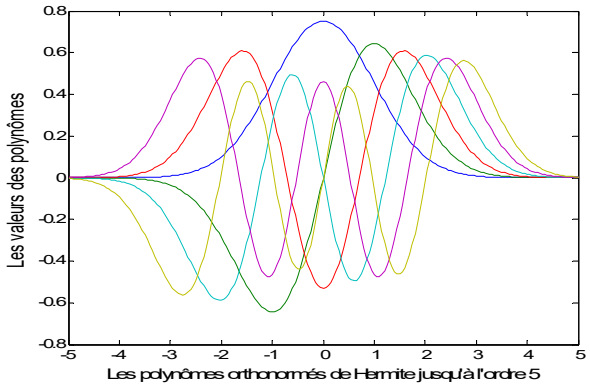


Figure (I.8): Courbes des polynômes orthogonaux d'Hermite pour les six premiers ordres.

2.1.7. Les polynômes de Bessel

Les polynômes orthogonaux de Bessel satisfont l'équation différentielle (I.26) pour

$$\sigma(x) = (x-a)^2 \ , \ \tau(x) = (\alpha+2)(x-a) + \delta, \ w(x) = (x-a)^{\alpha} e^{-\frac{\delta}{x-a}} \quad \text{et} \quad \lambda_n = 2n$$
$$\text{pour} \quad (a,\alpha) \in R^2 \ \text{et} \ \delta \succ 0 \tag{I.63}$$

Les polynômes orthogonaux de Bessel $B_n^{(a,\alpha,\delta)}(x)$ sont définis à partir de la formule de Rodriguez par la relation suivante [Chihara, 1978]:

$$B_n^{(a,\alpha,\delta)}(x) = \frac{(x-a)^{-\alpha} e^{\frac{\delta}{x-a}}}{(n+\alpha+1)_n} \frac{d^n}{dx^n}\left[(x-a)^{2n+\alpha} e^{-\frac{\delta}{x-a}}\right], \qquad |x-a| = 1 \tag{I.64}$$

En effectuant la dérivée nième du produit, nous obtenons une formule explicite des polynômes de Bessel $B_n^{(a,\alpha,\delta)}(x)$:

$$B_n^{(a,\alpha,\delta)}(x)=\sum_{k=0}^{n}\frac{n!\delta^{n-k}}{(n-k)!}\frac{\Gamma(n+k+\alpha+1)}{\Gamma(2n+\alpha+1)}\left(x-a\right)^k \tag{I.65}$$

Les polynômes de Bessel d'ordre n, sont définis à partir de la fonction hypergéométrique ${}_2F_0$ par :

$$B_n^{(a,\alpha,\delta)}(x)=\frac{\delta^n}{(n+\alpha+1)_n}\,{}_2F_0\left(\begin{matrix}-n,n+\alpha+1\\ -\end{matrix}\middle|-\frac{x-a}{\delta}\right) \tag{I.66}$$

Les polynômes de Bessel sont orthogonaux sur le cercle de centre a et de rayon 1 ($|x-a|=1$) relativement à la fonction de poids $w(x)=(x-a)^{\alpha}e^{-\frac{\delta}{x-a}}$.

$$\int_0^{\infty}B_n^{(a,\alpha,\delta)}(x)B_m^{(a,\alpha,\delta)}(x)(x-a)^{\alpha}e^{-\frac{\delta}{x-a}}dx=\delta_{n,m} \tag{I.67}$$

Les polynômes normalisés de Bessel sont définis par :

$$\tilde{B}_n^{(a,\alpha,\delta)}(x)=\sqrt{\frac{w(x)}{d_n^2}}B_n^{(a,\alpha,\delta)}(x) \tag{I.68}$$

Le calcul des polynômes de Bessel basé sur les fonctions hypergéométriques est très coûteux en temps de calcul. Pour remédier à ce problème nous utilisons la relation de récurrence des polynômes normalisés de Bessel donnée par :

$$\begin{aligned}&B_{n+1}^{(a,\alpha,\delta)}(x)=(x-\beta_n)B_n^{(a,\alpha,\delta)}(x)-\gamma_nB_{n-1}^{(a,\alpha,\delta)}(x),\qquad n\geq 1\\ &où\quad B_0^{(a,\alpha,\delta)}(x)=1\quad et\quad B_1^{(a,\alpha,\delta)}(x)=x-a+\frac{\delta}{\alpha+2}\end{aligned} \tag{I.69}$$

avec

$$\beta_n=\frac{4an(n+\alpha+1)+\alpha(a(\alpha+2)-\delta)}{(2n+\alpha)(2n+\alpha+2)}\ et\ \gamma_n=\frac{-n(n+\alpha)\delta^2}{(2n+\alpha-1)(2n+\alpha)^2(2n+\alpha+1)} \tag{I.70}$$

La figure (I.9) représente les courbes des six premiers ordres des polynômes orthonormés de Bessel.

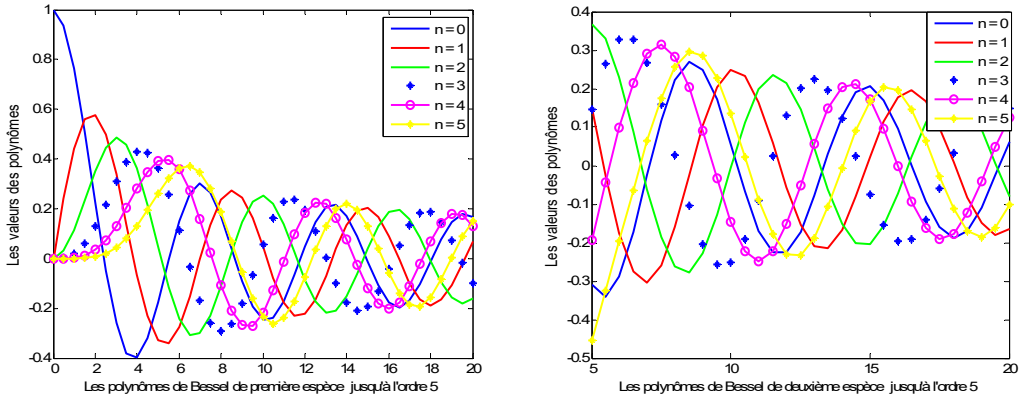


Figure (I.9): Courbes des polynômes orthogonaux de Bessel de première espèce et deuxième espèce pour les six premiers ordres.

2.2. Les moments orthogonaux continus

Les moments orthogonaux continus sont obtenus à partir de la projection de l'espace des données sur une base orthogonale d'une variable continue (Jacobi, Legendre, Chebyshev, Zernike, Gegenbauer, Fourier-Mellin, Bessel,….). Ces derniers possèdent les propriétés de robustesse contre le bruit et le pouvoir d'expression [Teh & Chin, 1988] qui assurent une représentation de l'image avec un minimum de redondance par rapport aux moments géométriques. Ces propriétés rendent les moments orthogonaux continus des bons candidats pour le traitement, l'analyse et la reconnaissance d'image par rapport aux moments géométriques [Teague, 1980].

Dans cette section, nous présentons les moments orthogonaux continus sur un rectangle définis à partir des polynômes de Jacobi, Legendre, Tchebychef, Gegenbauer et les moments orthogonaux continus sur un disque définis à partir des polynômes Zernike et Fourier-Mellin. Nous allons ainsi donner quelques propriétés relatives aux calculs des moments orthogonaux continus. Ensuite nous mettrons l'accent sur quelques problèmes limitant l'utilisation des moments orthogonaux continus en soulignant la nécessité d'utilisation des moments orthogonaux discrets.

2.2.1. Les moments orthogonaux continus sur un rectangle

Les moments orthogonaux continus sur un rectangle sont obtenus à partir de la projection de l'image sur une base de polynômes orthogonaux d'une variable continue sur un intervalle. Ces polynômes satisfont :

La relation d'orthogonalité:

$$\iint_D P_{pq}(x,y)P_{mn}(x,y)dxdy = 0 \qquad \textbf{(I.71)}$$

Où la relation d'orthogonalité pondérée :

$$\iint_D w(x,y)P_{pq}(x,y)P_{mn}(x,y)dxdy = 0 \qquad \textbf{(I.72)}$$

avec $p \neq m$ et $q \neq n$ et D est le domaine d'orthogonalité des polynômes $P_{pq}(x,y)$.

Les polynômes orthogonaux bidimensionnels $P_{pq}(x,y)$ sur un rectangle sont construits à partir des produits des deux polynômes $P_p(x)$ et $P_q(y)$.

Les moments orthogonaux continus sur le rectangle D sont définis par la formule suivante :

$$MOC_{pq} = \rho_p \rho_q \iint_D P_p(x)P_p(x)f(x,y)dxdy \;\; ; \;\; p,q = 0,1,2,..., \qquad \textbf{(I.73)}$$

avec ρ_p et ρ_q sont les facteurs de normalisation.

L'image $f(x,y)$ devrait être échantillonnée de telle sorte que son support soit contenu dans le domaine d'orthogonalité D. Les polynômes $P_p(x)$ satisfont la relation d'orthogonalité suivante :

$$\int_{D_1} w(x)P_p(x)P_q(x)dx = h_p \delta_{pq} \qquad \textbf{(I.74)}$$

où $w(x)$ est une fonction de poids, $h_p = \|P_p\|$ est la fonction norme carrée, D_1 est domaine d'orthogonalité des polynômes $P_p(x)$ tel que $D = D_1 \times D_2$ et $\delta_{p,q}$ est le symbole de Kronecker défini par :

$$\delta_{n,m} = \begin{cases} 1 & \text{si} \quad n = m \\ 0 & \text{si} \quad n \neq m \end{cases} \tag{I.75}$$

2.2.2. Les moments de Legendre

Les moments de Legendre (ML) [Teh & Chin, 1988] d'ordre (p+q) d'une image d'intensité $f(x,y)$ sont définis par :

$$ML_{p,q} = \frac{(2p+1)(2q+1)}{4} \int_{-1}^{1}\int_{-1}^{1} P_p(x) P_q(y) f(x,y) dxdy \tag{I.76}$$

où $p,q=0,1,2,...,\infty$

Les polynômes de Legendre $P_p(x)$ forment une base complète et orthogonale dans l'intervalle $[-1,1]$:

$$\int_{-1}^{1} P_p(x) P_q(y) = \frac{2}{2p+1} \delta_{pq} \tag{I.77}$$

Les moments de Legendre sont exprimés en fonction des moments géométriques par la relation suivante :

$$ML_{p,q} = \frac{(2p+1)(2q+1)}{4} \sum_{i=0}^{p} \sum_{j=0}^{q} a_{p,i} a_{q,j} MG_{i,j} \tag{I.78}$$

avec $a_{n,i}$ sont les coefficients des monômes de polynômes de Legendre $P_n(x)$ tel que $P_n(x) = \sum_{i=0}^{n} a_{n,i} x^i$. Ces coefficients sont définis par la relation suivante [Nikiforov & Uvarov, 1988] :

$$a_{n,i} = \begin{cases} (-1)^{(n-i)/2} \dfrac{1}{2^n} \dfrac{(n+i)!}{((n-i)/2)!((n+i)/2)!i!} & \text{si} \quad n\text{-}i \quad \text{pair} \\ 0 & \text{si} \quad n\text{-}i \quad \text{impair} \end{cases} \qquad \textbf{(I.79)}$$

Les quatre premiers moments de Legendre exprimés en termes des moments géométriques sont :

$$\begin{aligned} ML_{00} &= \frac{1}{4} MG_{00} \\ ML_{10} &= \frac{3}{4} MG_{10} \\ ML_{20} &= \frac{5}{8} (3MG_{20} - MG_{00}) \\ ML_{11} &= \frac{9}{4} MG_{11} \end{aligned} \qquad \textbf{(I.80)}$$

Si l'on considère le cas discret de la fonction intensité d'image donnant lieu aux échantillons $f(x_i, y_j)$ répartis selon une maille rectangulaire de taille (M, N) de pixels, alors l'équation (I.76) représentant les moments de la fonction continue f(x,y) peut être approchée en termes de sommation par la formule [Liao & Pawlak, 1996] :

$$ML_{p,q} = \frac{(2p+1)(2q+1)}{4} \sum_{i=1}^{M} \sum_{j=1}^{N} P_p(x_i) P_q(y_j) f(x_i, y_j) \Delta x \Delta y \qquad \textbf{(I.81)}$$

avec $x_i = -1 + \left(i + \frac{1}{2} \right) \Delta x$ et $y_i = -1 + \left(j + \frac{1}{2} \right) \Delta y$.

et $\Delta x = x_i - x_{i-1} = \frac{2}{M} =$ et $\Delta y = y_j - y_{j-1} = \frac{2}{N}$ sont les intervalles d'échantillonnage suivant les directions x et y.

2.2.3. Les moments de Chebyshev

Les moments de Chebyshev ont été introduits dans le traitement de l'image par Mukundan, qui a publié une série d'articles dans ce domaine [Mukundan & Ramakrishnan, 1998]. Les moments de Chebyshev (MCh) d'ordre $(p+q)$ d'image d'intensité $f(x,y)$ sont définis par :

$$MCh_{p,q} = \int_{-1}^{1}\int_{-1}^{1} T_p(x)T_q(y)f(x,y)dxdy \quad ; \; p,q = 0,1,2,.... \tag{I.82}$$

où $T_p(x)$ sont les polynômes de Chebyshev définis à partir des fonctions hypergéométriques par les relations suivantes [Nikiforov et al., 1991] :

$$\begin{aligned} T_n(x) &= {}_2F_1\left(-n,n;\frac{1}{2};\frac{1-x}{2}\right) \\ &= \frac{n}{2}\sum_{m=0}^{[n/2]}(-1)^m \frac{2^{n-2m}(n-m-1)!}{m!(n-2m)!}x^{n-2m} \\ &= \sum_{m=0}^{n} a_{n,k}x^k \end{aligned} \tag{I.83}$$

Les polynômes de Chebyshev de première espèce et de deuxième espèce sont définis en détail dans le chapitre I de cette thèse.

Les polynômes de Chebyshev $T_p(x)$ forment une base complète et orthogonale dans l'intervalle $[-1,1]$:

$$\int_{-1}^{1} T_p(x)T_q(y)\frac{1}{\sqrt{1-x^2}}dx = \begin{cases} 0 & : n \neq m \\ \pi & : n = m = 0 \\ \pi/2 & : \; n = m \neq 0 \end{cases} \tag{I.84}$$

Les moments de Chebyshev peuvent être exprimés en fonction des moments géométriques par la relation suivante :

$$MCh_{p,q} = \sum_{i=0}^{p}\sum_{j=0}^{q} a_{p,i}a_{q,j}MG_{i,j} \quad ; \; p,q = 0,1,2,... \tag{I.85}$$

avec $a_{n,i}$ sont les coefficients des monômes de polynômes de Chebyshev $T_n(x)$ définis par :

$$a_{n,n-2m} = \begin{cases} (-1)^m \dfrac{n}{2}\dfrac{(n-m-1)!}{m!(n-2m)!}2^{2-2m} & \text{si} \quad n \neq 0 \\ 0 & \text{si} \quad m \prec 0 \;\; \text{ou} \;\; m \succ n/2 \\ 1 & \text{si} \;\; n \neq 0 \end{cases} \tag{I.86}$$

Une propriété importante des moments de Tchebychev, qui est particulièrement utile dans le traitement de l'image, c'est qu'ils peuvent être modifiés pour préserver l'orthogonalité sur les images discrètes.

Dans le cas discret d'une image numérique de taille $M \times N$ pixels, l'équation (I.82) représentant les moments de Tchebychev de la fonction continue $f(x,y)$ peut être approchée en terme de sommation par la formule :

$$\begin{aligned} MCh_{p,q} &= \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} T_p(-1+2\frac{x}{M}) T_q(-1+2\frac{y}{N}) f(x,y)\Delta x \Delta y \\ &= \frac{4}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} T_p(-1+2\frac{x}{M}) T_q(-1+2\frac{y}{N}) f(x,y) \end{aligned} \quad \textbf{(I.87)}$$

2.2.4. Les moments de Gegenbauer

Les moments de Gegenbauer ont été introduits dans l'analyse d'image par Hosny [Hosny, 2011]. Les moments orthogonaux continus de Gegenbauer (MGe) d'ordre (p+q) d'une image d'intensité $f(x,y)$ sont définis par :

$$MGe_{p,q} = \frac{1}{\rho_p(\lambda)\rho_q(\lambda)} \int_{-1}^{1}\int_{-1}^{1} w^{(\lambda)}(x) w^{(\lambda)}(y) C_p^{(\lambda)}(x) C_q^{(\lambda)}(y) f(x,y) dxdy \quad ;\ p,q = 0,1,2,.... \quad \textbf{(I.88)}$$

où $\rho_p(\lambda)$ est la fonction de poids, $w^{(\lambda)}(x)$ est le facteur de normalisation et $C_p^{(\lambda)}(x)$ sont les polynômes de Gegenbauer définis à partir des fonctions hypergéométriques par les relations suivantes :

$$\begin{aligned} C_n^{(\lambda)}(x) &= \frac{(2\lambda)_n}{n!} {}_2F_1\left(-n, n+2\lambda; \lambda+\frac{1}{2}; \frac{1-x}{2}\right) \\ &= \frac{1}{\Gamma(\lambda)} \sum_{m=0}^{[n/2]} (-1)^m \frac{2^{n-2m}\Gamma(\lambda+n-m)}{m!(n-2m)!} x^{n-2m} \\ &= \sum_{m=0}^{n} a_{n,k} x^k \end{aligned} \quad \textbf{(I.89)}$$

Ces polynômes sont définis en détail dans le chapitre I de cette thèse.

Les polynômes de Gegenbauer $C_p^{(\lambda)}(x)$ forment une base orthogonale dans l'intervalle $[-1,1]$:

$$\int_{-1}^{1} w^{(\lambda)}(x) C_n^{(\lambda)}(x) C_m^{(\lambda)}(x) dx = \rho_n(\lambda)\delta_{nm} \tag{I.90}$$

avec

$$w^{(\lambda)}(x) = (1-x^2)^{\lambda-\frac{1}{2}}$$

et

$$\rho_n(\lambda) = \begin{cases} \dfrac{\pi 2^{1-2\lambda}\Gamma(n+2\lambda)}{(n+\lambda)n!\Gamma^2(\lambda)} & : \lambda \neq 0 \\ \dfrac{2\pi}{n^2} & : \lambda = 0 \end{cases} \tag{I.91}$$

Les moments de Gegenbauer sont exprimés en fonction des moments géométriques par la relation suivante :

$$MGe_{p,q} = \sum_{i=0}^{p}\sum_{j=0}^{q} a_{p,i} a_{q,j} MG_{i,j} \quad ; \; p,q = 0,1,2,... \tag{I.92}$$

avec $a_{n,i}$ sont les coefficients des monômes de polynômes de Gegenbauer $C_p^{(\lambda)}(x)$ définis par la relation suivante :

$$a_{n,n-2m} = \begin{cases} (-1)^m \dfrac{n}{2} \dfrac{(n-m-1)!}{m!(n-2m)!} 2^{2-2m} & \text{si} \quad n \neq 0 \\ 0 & \text{si} \quad m \prec 0 \;\text{ ou }\; m \succ n/2 \\ 1 & \text{si} \;\; n \neq 0 \end{cases} \tag{I.93}$$

Pour une image numérique $f(x,y)$ de taille $M \times N$, l'équation (I.88) représentant les moments de Gegenbauer peut être approchée en terme de sommation par la formule [Hosny, 2011] :

$$MGe_{p,q} = \frac{1}{\rho_p(\lambda)\rho_q(\lambda)} \sum_{i=0}^{M-1}\sum_{j=0}^{N-1} w^{(\lambda)}(x_i) w^{(\lambda)}(y_j) C_p^{(\lambda)}(x_i) C_q^{(\lambda)}(y_j) f(i,j) \Delta x \Delta y$$
$$= \frac{4}{MN} \times \frac{1}{\rho_p(\lambda)\rho_q(\lambda)} \sum_{i=0}^{M-1}\sum_{j=0}^{N-1} w^{(\lambda)}(x_i) w^{(\lambda)}(y_j) C_p^{(\lambda)}(x_i) C_q^{(\lambda)}(y_j) f(i,j) \quad \textbf{(I.94)}$$

avec

$$x_i = -1 + 2\frac{i}{M} \quad ; \quad y_j = -1 + 2\frac{j}{N}$$
$$\Delta x = x_{i+} - x_i \quad ; \quad \Delta y = y_{j+1} - y_j$$

2.2.5. Les moments de Gauss-Hermite

Les moments orthogonaux continus de Gauss-Hermite ont été introduits pour la première fois dans le domaine d'image par Shen en 1997 [Shen, 1997]. Ils sont largement utilisés par des autres chercheurs dans les années qui suivent [Hosny, 2012; Wu & Shen, 2005 ; Hmimid et al., 2012]. Les moments de Gauss-Hermite (MGH) d'ordre $(p+q)$ d'une image d'intensité $f(x,y)$ sont définis par :

$$MGH_{p,q} = \frac{1}{\rho_p \rho_q} \int_{-1}^{1}\int_{-1}^{1} w(x) w(y) H_p(x) H_q(y) f(x,y) dxdy \quad ; \; p,q = 0,1,2,.... \quad \textbf{(I.95)}$$

où ρ_p est le facteur de normalisation, $w(x)$ est la fonction de poids et $H_p(x)$ sont les polynômes de Gauss-Hermite définis à partir des fonctions hypergéométriques par les relations suivantes :

$$H_n(x) = {}_2F_0\left(\frac{-n}{2}, -\frac{n-1}{2}; -; \frac{-1}{x^2}\right)$$
$$= \sum_{m=0}^{[n/2]} (-1)^m \frac{2^{n-2m} n!}{m!(n-2m)!} x^{n-2m}$$
$$= \sum_{k=0}^{n} a_{n,k} x^k \quad \textbf{(I.96)}$$

Les polynômes de Gauss-Hermite $H_p(x)$ sont définis en détail dans [Sayyour, 2012]. Ils sont orthogonaux dans l'intervalle $[-1,1]$:

$$\int_{-1}^{1} w(x)H_n(x)H_m(x)dx = \rho_n \delta_{nm} \tag{I.97}$$

avec $w(x) = e^{-x^2}$ et $\rho_n = 2^n n! \sqrt{\pi}$

Les polynômes orthogonaux normalisés de Gauss-Hermite sont définis par :

$$\hat{H}_n(x) = \sqrt{\frac{w(x)}{\rho_n}} H_n(x) \tag{I.98}$$

Pour une image numérique $f(x,y)$ de taille $M \times N$ pixels, les moments orthogonaux de Gauss-Hermite sont approchés en terme de sommation par la formule [Hosny, 2012] :

$$\begin{aligned} MGH_{p,q} &= \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} \hat{H}_p(-1+2\frac{x}{M})\hat{H}_q(-1+2\frac{y}{N}) f(x,y)\Delta x \Delta y \\ &= \frac{4}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} \hat{H}_p(-1+2\frac{x}{M})\hat{H}_q(-1+2\frac{y}{N}) f(x,y) \end{aligned} \tag{I.99}$$

2.2.6. Les moments continus orthogonaux sur un disque

Les moments orthogonaux continus sur un disque sont obtenus à partir de la projection de l'image sur une base de polynômes orthogonaux d'une variable continue sur un disque. L'une des principales motivations quand on travaille avec les moments orthogonaux sur un disque (Zernike, Pseudo-Zernike, Bessel, Fourier-Mellin,…) est la construction facile des moments invariants à la rotation. Ces moments sont généralement construits sous la forme:

$$\upsilon_{pq} = n_{pq} \int_0^{2\pi} \int_0^1 R_{pq}(r) e^{-iq\varphi} f(r,\varphi) dr d\varphi \quad ; \quad p = 0,1,2,... \; ; \; q = -p,...,p \tag{I.100}$$

Avec :

n_{pq} : Le facteur de normalisation

r, φ : Les coordonnées polaires

$R_{pq}(r)$: La partie radiale dans le polynôme

$e^{-iq\varphi}$: La partie angulaire dans le polynôme

$f(r,\varphi)$: L'intensité de l'image dans le disque

Le domaine d'orthogonalité est le disque unité $D = \{(x, y) / x^2 + y^2 \leq 1\}$

Pour calculer les moments orthogonaux sur le disque unité, l'image $f(x, y)$ doit être transformée de telle sorte qu'elle soit entièrement contenue dans le disque.

2.2.7. Les moments de Zernike

Les moments de Zernike sont fréquemment utilisés pour capturer les caractéristiques globales d'une image en reconnaissance et en analyse d'images. Introduit pour la première fois en vision par ordinateur par Teague [Teague, 1980], ce descripteur de forme a prouvé sa supériorité sur les autres fonctions de moments [Belkasim et al., 1991; Liao & Pawlak, 1996] par sa capacité de description et sa robustesse au bruit et aux déformations. Teague a montré que les moments invariants de Zernike du deuxième et troisième ordre sont équivalents aux invariants de Hu lorsqu'ils sont exprimés en fonctions de moments géométriques. Il a présenté également les invariants jusqu'au huitième ordre sous forme explicite mais il n'a pas donné de règle générale pour les obtenir [Teague, 1980]. Plus tard, Wallin & Kübler [Wallin & Kübler, 1995] ont décrit un algorithme pour la construction des invariants à la rotation pour n'importe quel ordre. Très récemment encore, de nombreux chercheurs se sont penchés sur ces moments, principalement pour optimiser leur temps de calcul [Chong et al., 2003; Wee & Paramesran, 2006; Wee & Paramesran, 2007 ; Kotoulas & Andreadis, 2005] et améliorer leur précision [Amayeh et al., 2005].

Les fonctions complexes de Zernike constituent un ensemble de fonctions orthogonales définies sur le disque unité. Les moments de Zernike d'une image sont construits par la projection de cette image sur ces fonctions. Elles possèdent trois propriétés principales : l'orthogonalité, l'invariance en rotation et le compactage de l'information.

Mathématiquement, les moments de Zernike sont définis pour un ordre p et une répétition q sur le disque unité $x^2+y^2\leq 1$ par la relation suivante :

$$A_{p,q} = \frac{(p+1)}{\pi} \iint_{x^2+y^2\leq 1} f(x,y) V^*_{p,q}(\rho,\phi)\, dx\, dy \qquad \textbf{(I.101)}$$

La fonction de Zernike [Zernike, 1934] d'ordre (p,q) est définie sur le disque unité $x^2+y^2\le 1$ (noté D) par :

$$V_{p,q}(\rho,\varphi)=R_{p,q}(\rho)\exp(jq\varphi) \tag{I.102}$$

où $p=0,1,\ldots,\infty$

q : prend des valeurs positives ou négatives telles que $p-|q|$ pair et $|q|\le p$.

ρ: Longueur du vecteur à partir de l'origine.

ϕ : Angle entre ρ et l'axe des *x*.

$V^*_{p,q}(\rho,\varphi)$ Dénote le conjugué complexe de $V_{p,q}(\rho,\varphi)$

La forme radiale des polynômes de Zernike est donnée par :

$$R_{p,q}(\rho)=\sum_{s=0}^{(p-|q|)/2}\frac{(-1)^s\,(p-s)!}{s!\left(\frac{p+|q|}{2}-s\right)!\left(\frac{p-|q|}{2}-s\right)!}\rho^{p-2s}=\sum_{k=|q|,|q|+2,\ldots}^{p}B_{pqk}\rho^k \tag{I.103}$$

avec

$$B_{pqk}=\frac{(-1)^{(p-k)/2}\,((p+k)/2)!}{((p-k)/2)!((k+q)/2)!((k-q)/2)!} \tag{I.104}$$

Ces polynômes sont orthogonaux et satisfont à la relation:

$$\iint_D V^*_{p,q}(x,y)V_{p',q'}(x,y)\,dx\,dy=\frac{\pi}{(p+1)}\delta_{p,p'}\delta_{q,q'} \tag{I.105}$$

En particulier

$$\int_0^1 R_{p,q}(\rho)\,R_{p',q}(\rho)\,\rho d\rho=\frac{1}{2(p+1)}\;\delta_{p,p'} \tag{I.106}$$

La formule de récurrence pour le calcul de la partie radiale des polynômes de Zernike est :

$$R_{p,q}(\rho)=\frac{2\rho p}{p+q}R_{p-1,q-1}(\rho)-\frac{p-q}{p+q}R_{p-2,q}(\rho) \tag{I.107}$$

Les moments de Zernike peuvent être exprimés en fonction des moments géométriques par la relation suivante :

$$MZ_{p,q} = \frac{(p+1)}{\pi} \sum_{k=|q|,|q|+2,\ldots}^{p} \sum_{j=0}^{(k-|q|)/2} \sum_{m=0}^{|q|} \binom{(k-|q|)/2}{j} \binom{|q|}{m} w^m B_{pqk} MGk-2j-m, 2j+m \quad \textbf{(I.108)}$$

avec

$$w = \begin{cases} -i & q \geq 0 \\ i & q \leq 0 \end{cases} \quad \textbf{(I.109)}$$

Le calcul des polynômes de Zernike par cette formule doit commencer par

$$R_{p,p}(\rho) = \rho^p, R_{p,-p}(\rho) = \rho^p, p = 0,1,.... \quad \textbf{(I.110)}$$

puis les polynômes de Zernike peuvent être calculés pour $p = 2,3,....$ et $q = -p+2, -p+4, ..., p-2$.

Des exemples de polynômes radiaux :

$$\begin{aligned} R_{00}(\rho) &= 1 \\ R_{11}(\rho) &= \rho \\ R_{20}(\rho) &= 2\rho^2 - 1 \\ R_{22}(\rho) &= \rho^2 \\ R_{31}(\rho) &= 3\rho^3 - 2\rho \\ R_{33}(\rho) &= \rho^3 \end{aligned} \quad \textbf{(I.111)}$$

La fonction radiale est symétrique par rapport à la répétition q, à savoir $R_{p,-q}(\rho) = R_{p,q}(\rho)$

Les graphes des polynômes radiaux sont représentés sur la Figure (I.10).

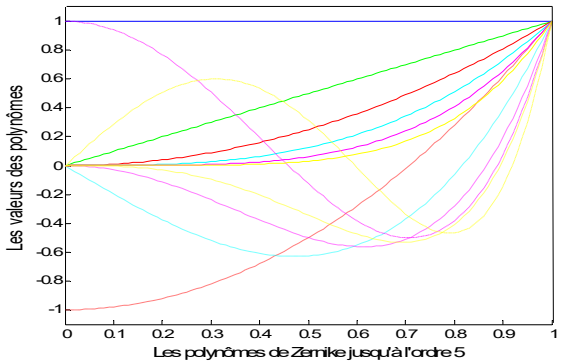


FIGURE (I.10): Courbes des polynômes radiaux de Zernike pour les six premiers ordres.

2.2.8. Les moments orthogonaux de Fourier-Mellin

Les moments orthogonaux de Fourier-Mellin (MFM) ont été proposés par Sheng et Shen dans [Sheng & Shen, 1994].et par Zhang et al dans [Zhang et al, 2010]. Ils sont définis par la relation suivante :

$$MFM_{nl} = \frac{(n+1)}{\pi}\int_0^{2\pi}\int_0^1 Q_n(r)e^{-il\varphi}f(r,\varphi)rdrd\varphi \qquad n=0,\pm1,\pm2,..., \tag{I.112}$$

Où $Q_n(r)$ est la fonction radiale des polynômes de Fourier-Mellin définie par :

$$Q_n(r) = \sum_{s=0}^{n}(-1)^{n+s}\frac{(n+s+1)!}{(n-s)!s!(s+1)!}r^s \tag{I.113}$$

Cette fonction est définie en fonction des polynômes de Jacobi par la relation suivante :

$$Q_n(r) = (-1)^n\binom{n+1}{n}J_n^{(2,2)}(r) \tag{I.114}$$

La relation d'orthogonalité des fonctions radiales est donnée par :

$$\int_0^1 Q_n(r)Q_m(r)rdr = \frac{1}{2(n+1)}\delta_{nm} \tag{I.115}$$

Les cinq premiers polynômes radiaux sont définis par les relations suivantes :

$$
\begin{aligned}
Q_0(r) &= 1 \\
Q_1(r) &= -2 + 3r \\
Q_2(r) &= 3 - 12r + 10r^2 \\
Q_3(r) &= -4 + 30r - 60r^2 + 35r^3 \\
Q_4(r) &= 5 - 60r + 210r^2 - 280r^3 + 126r^4
\end{aligned}
\qquad \textbf{(I.116)}
$$

La Figure (I.11) représente les graphes des six premiers polynômes radiaux de Fourier-Mellin.

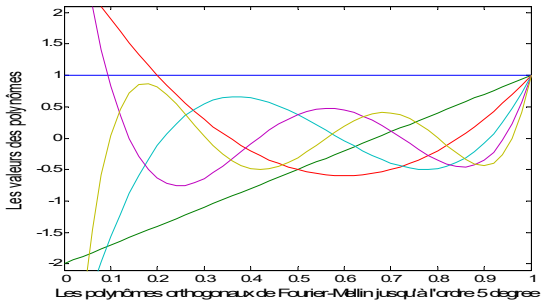


FIGURE (I.11): Courbes des polynômes orthogonaux de Fourier-Mellin pour les six premiers ordres.

Le nom de "moments de Fourier-Mellin" est associé à leur relation à la transformée Fourier-Mellin pour les entiers s et l :

$$
M_f(s,l) = \frac{1}{2\pi}\int_0^{\infty}\int_0^{2\pi} r^s e^{-il\varphi} f(r,\varphi)\, d\varphi \frac{dr}{r}
\qquad \textbf{(I.117)}
$$

La relation entre ces deux grandeurs est :

$$
MFM_{nl} = 2(n+1)\sum_{s=0}^{n}(-1)^{n+s}\frac{(n+s+1)!}{(n-s)!\,s!\,(s+1)!} M_f(s+2,l)
\qquad \textbf{(I.118)}
$$

La principale différence entre les moments orthogonaux Fourier-Mellin et les moments de Zernike est l'absence de la contrainte $n \le l$ et l'indépendance de la fonction radiale $Q_n(r)$ du facteur de répétition. La répartition différente des zéros sur le disque unité est également intéressante. Il existe une relation directe ente la fonction radiale de Fourier-Mellin et la fonction radiale Zernike :

$$
rQ_n(r^2) = R_{2n+1,1}(r)
\qquad \textbf{(I.119)}
$$

Les moments orthogonaux Fourier-Mellin peuvent être exprimés en fonction des moments complexes :

$$MFM_{nl} = \frac{(n+1)}{\pi} \sum_{s=0}^{n} (-1)^{n+s} \frac{(n+s+1)!}{(n-s)!\,s!\,(s+1)!} MC_{(s-l)/2,(s+2)/l} \qquad \textbf{(I.120)}$$

L'utilisation des moments complexes pour calculer les moments de Fourier-Mellin nous permet d'éviter un calcul complexe des moments géométriques radiaux.

3. Polynômes et moments discrets

3.1. Polynômes orthogonaux d'une variable discrète

Les familles des polynômes de Tchebichef, Charlier, Mexnier, Krawchouk et Hahn sont appelées polynômes orthogonaux d'une variable discrète [Nikiforov & Uvarov, 1988 ; Nikiforov et al., 1991]. Elles forment une classe importante de fonction spéciale dans le traitement des divers problèmes de mathématiques, physique théorique et d'ingénierie. L'étude de ces polynômes a été initiée par P.L. Chebyshev au milieu du siècle dernier.

Les polynômes orthogonaux d'une variable discrète peuvent être caractérisés en termes d'équation de différence de type hypergéométrique suivante :

$$\sigma(x)\Delta\nabla P_n(x) + \tau(x)\Delta P_n(x) + \lambda_n P_n(x) = 0 \qquad \textbf{(I.121)}$$

où

- $\Delta P_n(x) = P_n(x+1) - P_n(x)$; $\nabla P_n(x) = P_n(x) - P_n(x-1)$ indiquent l'opérateur de différence en avance et l'opérateur de différence en arrière .
- $\sigma(x)$ et $\tau(x)$ sont des polynômes de deuxième et premier degré, respectivement.
- λ_n est une constante appropriée.

3.1.1. Forme générale des polynômes orthogonaux discrets

La forme générale des polynômes orthogonaux discrets est obtenue en résolvant l'équation de différence (I.121). Les solutions de cette équation s'expriment par la formule de Rodriguez suivante [Nikiforov et al., 1991]:

$$P_n(x) = \frac{B_n}{w(x)} \nabla^n [w_n(x)] \tag{I.122}$$

Où $w(x)$ est la fonction de poids de $w_n(x)$ dans le cas $n=0$.

Les polynômes solutions de l'équation (I.74) sont déterminés par la formule de Rodriguez (I.122) selon les facteurs de normalisation B_n.

En utilisant des opérations successives de l'opérateur différence en arrière, nous obtenons la propriété suivante [Nikiforov et al., 1991] :

$$\nabla^n f(x) = \sum_{k=0}^{n} \binom{n}{k} (-1)^k f(x-k) \tag{I.123}$$

En combinant les équations (I.122) et (I.123), nous obtenons l'expression explicite des polynômes orthogonaux d'une variable discrète :

$$P_n(x) = \frac{B_n}{w(x)} \sum_{k=0}^{n} \binom{n}{k} (-1)^k w_n(x-k) \tag{I.124}$$

3.1.2. Relation de récurrence par rapport à l'ordre n

La forme explicite des polynômes orthogonaux d'une variable discrète (I.124) nous permet de définir une relation de récurrence entre trois successifs P_{n+1}, P_n et P_{n-1} .cette relation est définie comme suit [Nikiforov et al., 1991] :

$$xP_n(x) = \alpha_n P_{n+1}(x) + \beta_n P_n(x) + \gamma_n P_{n-1}(x) \tag{I.125}$$

Où α_n, β_n et γ_n sont des coefficients donnés dans le tableau I.1.

3.1.3. Propriété d'orthogonalité des polynômes orthogonaux discrets

Les polynômes orthogonaux discrets d'une seule variable satisfont à la relation d'orthogonalité suivante :

$$\sum_{x=0}^{s} P_n(x)P_m(x)w(x) = d_n^2.\delta_{nm} \quad ; \ 0 \le m,n \le s \tag{I.126}$$

où d_n^2 représente la fonction norme carrée des polynômes orthogonaux correspondants et δ_{nm} désigne la fonction de Dirac. Ces fonctions sont données dans le tableau (I.1) pour chaque type de polynômes.

Les polynômes orthogonaux normalisés sont obtenus en utilisant la fonction norme carrée d_n^2 et la fonction de poids $w(x)$ par la relation suivante :

$$\tilde{P}_n(x) = P_n(x)\sqrt{\frac{w(x)}{d_n^2}} \ ; \quad \text{n=0,1,2...,s} \tag{I.127}$$

Par conséquent, la propriété d'orthogonalité des polynômes orthogonaux normalisés dans (I.126) devient :

$$\sum_{x=0}^{s} \tilde{P}_n(x)\tilde{P}_m(x) = \delta_{nm} \quad ; \ 0 \le m,n \le s \tag{I.128}$$

Les données de base pour les polynômes orthogonaux d'une variable discrète de Tchebichef, Charlier, Mexnier, Krawchouk et Hahn sont énumérées dans le tableau (I.1) [Nikiforov et al., 1991 ; Zhu et al., 2010].

3.1.4. Domaines d'orthogonalité des polynômes orthogonaux discrets

Le domaine d'orthogonalité des polynômes orthogonaux discrets dépend du domaine de définition des fonctions de poids $w(x)$. Ces dernières sont définies dans le domaine discret suivant :

$$G = \{x / 0 \le x \le s\} \tag{I.129}$$

où $s = N$ pour le cas des polynômes de Krawtchouk, $s = N-1$ pour le cas des polynômes de Tchebichef et Hahn et $s = \infty$ pour le cas des polynômes de Charlier et Meixner.

Tableau (I.1): Les données de base pour les polynômes orthogonaux discrets de Tchebichef, Krawtchouk, Hahn, Meixner et Charlier.

$P_n(x)$	Tchebichef $t_n(x;N)$	Krawtchouk $K_n(x;p,N)$	Charlier $C_n^{a_1}(x)$	Meixner $M_n^{(\beta,\mu)}(x)$	Hahn $h_n^{(a,b)}(x,N)$
s	$N-1$	N	∞	∞	$N-1$
$\sigma(x)$	$x(N-x)$	x	x	x	$x(N+a-x)$
$\tau(x)$	$N-1-2x$	$\frac{Np-x}{1-p}$	a_1-x	$\beta\mu-x(1-\mu)$	$(b+1)(N-1)-(a+b+2)x$
λ_n	$n(n+1)$	$\frac{n}{1-p}$	n	$n(1-\mu)$	$n(a+b+n+1)$
B_n	$\frac{(-1)^n}{n!}$	$\frac{(-1)^n(1-p)^n}{n!}$	$\frac{1}{a_1^n}$	$\frac{1}{\mu^n}$	$\frac{(-1)^n}{n!}$
$w_n(x)$	$\frac{\Gamma(N-x)\Gamma(n+1+x)}{\Gamma(N-n-x)\Gamma(x+1)}$	$\frac{N!p^{x+N}(1-p)^{N-n-x}}{\Gamma(x+1)\Gamma(N+1-n-x)}$	$\frac{e^{-a_1}a_1^{x+n}}{x!}$	$\frac{\mu^{x+n}\Gamma(n+\beta+x}{\Gamma(\beta)x!}$	$\frac{\Gamma(N+a-x)\Gamma(n+b+1+x)}{\Gamma(N-n-x)\Gamma(x+1)}$
d_n^2	$\frac{(N+n)!}{(2n+1)(N-n-1)!}$	$\frac{N!(p(1-p))^n}{n!(N-n)!}$	$\frac{n!}{a_1^n}$	$\frac{n!(\beta)_n}{\mu^n(1-\mu)^\delta}$	$\frac{\Gamma(a+n+1)\Gamma(b+n+1)(a+b+n+1)_N}{(a+b+2n+1)n!(N-n-1)}$
α_n	$\frac{n+1}{2(2n+1)}$	$n+1$	$-a_1$	$\frac{\mu}{1-\mu}$	$\frac{(n+1)(a+b+n+1)}{(a+b+2n+1)n!(N-n-1)}$
β_n	$\frac{N-1}{2}$	$n+p(N-2n)$	$n+a_1$	$\frac{n+\mu(n+\beta)}{1-\mu}$	$\frac{a-b+2N-2}{4}+\frac{(b-a)(a+b+2N)}{4(a+b+2n)(a+b+2n+2)}$
γ_n	$\frac{n(N^2-n^2)}{2(2n+1)}$	$p(1-p)(N-n+1)$	$-n$	$\frac{n(n-1+\beta)}{\mu-1}$	$\frac{(a+n)(b+n)(a+b+N+n)(N-n)}{(a+b+2n)(a+b+2n+1)}$

3.1.5. Relation de récurrence des polynômes orthonormés discrets

La relation de récurrence de trois termes des polynômes orthonormés discrets $\tilde{P}_n(x)$ est obtenue à partir de la relation de récurrence de trois termes des polynômes orthogonaux discrets (I.125) et la forme orthonormé des polynômes discrets (I. 127) par la relation suivante [Koekoek et al., 2010] :

$$\tilde{P}_n(x)=\frac{B\times D}{A}\tilde{P}_{n-1}(x)-\frac{C\times E}{A}\tilde{P}_{n-2}(x) \qquad \textbf{(I.130)}$$

où A, B, C, D et E sont des coefficients calculés d'après la formule de Rodriguez (I.123) et les équations (I.124), et (I.25). Ces derniers sont donnés dans le tableau (I.2).

Tableau (I.2): Les données de la relation de récurrence à trois termes des polynômes orthogonaux discrets.

$\tilde{P}_n(x)$	**Tchebichef** $t_n(x;N)$	**Krawtchouk** $K_n(x;p,N)$	**Charlier** $C_n^{a_1}(x)$	**Meixner** $M_n^{(\beta,\mu)}(x)$	**Hahn** $h_n^{(a,b)}(x,N)$
A	$\frac{n}{2(2n-1)}$	n	$-a_1$	$\frac{\mu}{\mu-1}$	$\frac{n(a+b+n)}{(a+b+2n-1)(a+b+2n)}$
B	$x-\frac{N-1}{2}$	$x-n+1-p(N-2n+2)$	$x-n+1-a_1$	$\frac{x-x\mu-n+1-\mu n+\mu-\beta\mu}{1-\mu}$	$x-\frac{a-b+2N-2}{4}$ $-\frac{(b^2-a^2)(a+b+2N)}{4(a+b+2n-2)(a+b+2n)}$
C	$-\frac{(n-1)(N^2-(n-}{2(2n-1)}$	$-p(1-p)(N-n+2)$	$n-1$	$\frac{(n-1)(n-2+\beta)}{1-\mu}$	$-\frac{(a+n-1)(b+n-1)}{(a+b+2n-2)}$ $\times\frac{(a+b+N+n-1)(N-n+1)}{(a+b+2n-1)}$
D	$\sqrt{\frac{2n+1}{(N^2-n^2)(2n-}}$	$\sqrt{\frac{n}{p(1-p)(N-n+1)}}$	$\sqrt{\frac{a_1}{n}}$	$\sqrt{\frac{\mu}{n(\beta+n-1)}}$	$\sqrt{\frac{n(a+b+n)}{(N-n)(a+n)(b+n)}}$ $\times\sqrt{\frac{(a+b+2n+1)}{(a+b+2n-1)(a+b+n+N)}}$
E	$\sqrt{\frac{2n+1}{(N^2-(n-1)^2)}}$ $\times\sqrt{\frac{1}{(N^2-n^2)(2n-}}$	$\sqrt{\frac{n(n-1)}{(p(1-p))^2(N-n+2)}}$ $\times\sqrt{\frac{1}{(N-n+1)}}$	$\sqrt{\frac{a_1^2}{n(n-1)}}$	$\sqrt{\frac{\mu^2}{n(n-1)(\beta+n-2)}}$ $\times\sqrt{\frac{1}{(\beta+n-1)}}$	$\sqrt{\frac{n(n-1)(a+b+n)}{(a+n)(a+n-1)(b+n)(b+n-1)(N-n+1)(N-n)}}$ $\times\sqrt{\frac{(a+b+n-1)(a+b+2n+1)}{(a+b+2n-3)(a+b+n+N)(a+b+n+N-1)}}$

Après avoir défini, dans le paragraphe précèdent, la formule générale (I.83) des polynômes orthogonaux discrets, nous allons étudier en détail les différentes propriétés des polynômes de Tchebichef, krawtchouk, Hahn, Meixner et Charlier.

3.1.6. Les polynômes de Tchebichef

Les polynômes orthogonaux d'une variable discrète de Tchebichef $t_n(x;N)$ satisfont l'équation de différences partielles du second ordre suivante :

$$x(N-x)\Delta\nabla t_n(x;N)+(N-1-2x)\Delta t_n(x;N)+n(n+1)t_n(x;N)=0$$

avec n,x=0,1,2.....N-1 **(I.131)**

En utilisant la formule de Rodriguez (I.123), les polynômes orthogonaux de Tchebichef d'ordre n sont définis à partir de la fonction hypergéométrique ${}_3F_2$ comme suit :

$$t_n(x,N) = (1-N)_n \, {}_3F_2(-n,-x,1+n;1-N;1) \tag{I.132}$$

Les premiers ordres n=0 et n=1 des polynômes normalisés discrets de Tchebichef sont calculés à partir de l'équation (I.131) par les relations suivantes :

$$\tilde{t}_0(x;N) = \sqrt{1/N} \quad \text{et} \quad \tilde{t}_1(x;N) = (2x-N+1)\sqrt{\frac{3}{(N(N^2-1))}} \tag{I.133}$$

Les autres valeurs des polynômes de Tchebichef $2 \le n \le N-1$ sont déduites de la relation de récurrence par rapport à l'ordre n définie par l'équation (I.130) sachant que les coefficients A, B, C, D et E sont donnés dans le tableau (I.2).

La figure (I.12) représente les courbes des six premiers ordres des polynômes orthonormés de Tchebichef.

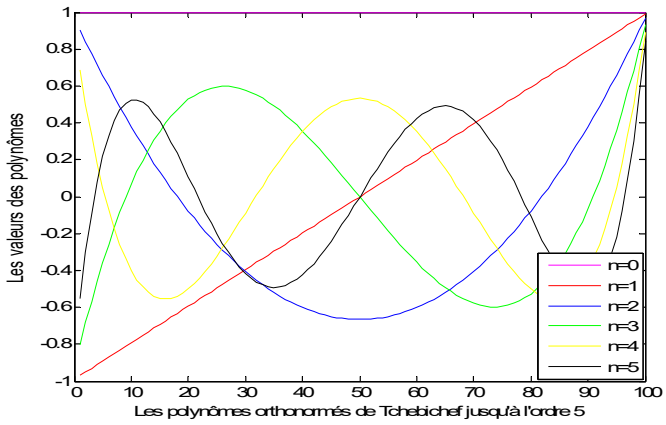


Figure (I.12): Courbes des polynômes orthonormés de Tchebichef pour les six premiers ordres.

3.1.7. Les polynômes de Krawtchouk

Les polynômes orthogonaux discrets de Krawtchouk $K_n(x; p, N)$ d'une seule variable satisfont à l'équation de différences partielles du second ordre suivante :

$$(1-p)x\Delta\nabla K_n(x;p,N)+(Np-x)\Delta K_n(x;p,N)+nK_n(x;p,N)=0$$
$$avec \quad x,n=0,1,2\ldots N \quad et \quad 0\prec p\prec 1 \tag{I.134}$$

Le nième polynôme de Krawtchouk est défini en utilisant la fonction hypergéométrique ${}_2F_1$ par la relation suivante :

$$k_n(x;p,N)={}_2F_1(-n,-x,-N;\frac{1}{p}) \tag{I.135}$$

Les deux premiers ordres n=0 et n=1 des polynômes normalisés discrets de Krawtchouk sont donnés par les relations suivantes.

$$\tilde{k}_0(x;p,N)=\sqrt{\frac{N!\,p^x(1-p)^{N-x}}{x!(N-x)!}}$$
$$\tilde{k}_1(x;p,N)=(-p(N-x)+x(1-p))\sqrt{\frac{(N-1)!\,p^{x-1}(1-p)^{N-x-1}}{x!(N-x)!}} \tag{I.136}$$

Les autres valeurs des polynômes de Krawtchouk $2\le n\le N$ sont déduites de la relation de récurrence par rapport à l'ordre n définie par l'équation (I.130) sachant que les coefficients A, B, C, D et E sont donnés dans le tableau (I.2).

La figure (I.13) représente les courbes des six premiers ordres des polynômes orthonormés de Krawtchouk avec trois valeurs différentes du paramètre p .

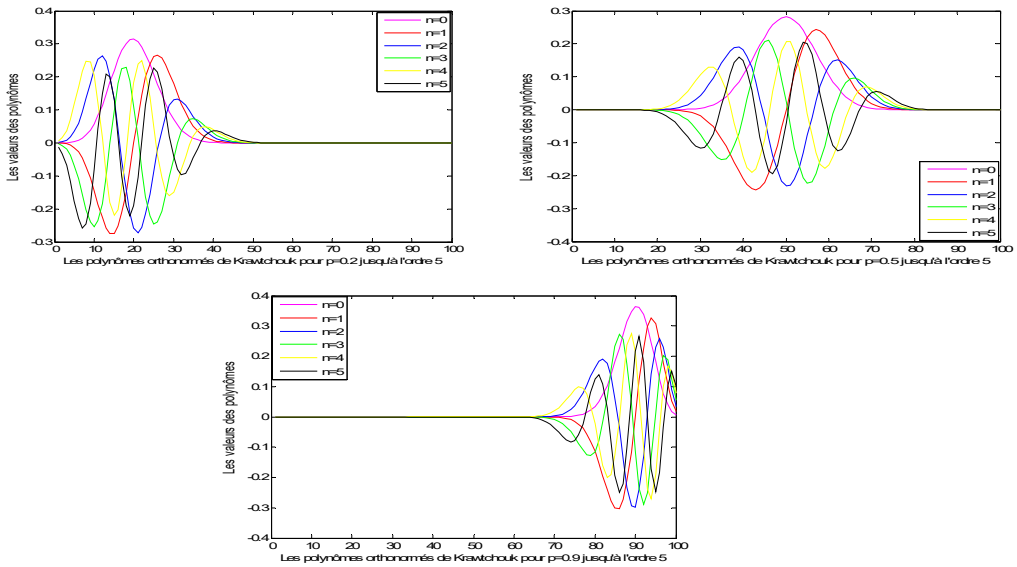


Figure (I.13): Courbes des polynômes orthonormés de Krawtchouk pour $p=0.2,\ p=0.5,$ et $p=0.9$ pour les six premiers ordres.

3.1.8. Les polynômes de Hahn

Les polynômes orthogonaux discrets de Hahn $h_n^{(a,b)}(x,N)$ d'une seule variable satisfont à l'équation de différences partielles du second ordre suivante :

$$x(N+a-x)\Delta\nabla h_n^{(a,b)}(x,N)+[(b+1)(N-1)-(a+b+2)x]\Delta h_n^{(a,b)}(x,N)+n(a+b+n+1)h_n^{(a,b)}(x,N)=0$$

avec $x,n=0,1,2\ldots N-1$, $a\succ 0$ *et* $b\succ 0$

(I.137)

En utilisant la formule de Rodriguez (I.123), on obtient la forme explicite des polynômes orthogonaux de Hahn d'ordre n en fonction de la fonction hypergéométrique ${}_3F_2$:

$$h_n^{(a,b)}(x,N)=\frac{(1)^n(b+1)_n(N-n)_n}{n!}\,{}_3F_2(-n,-x,n+1+a+b;b+1,1-N;1)$$

avec $n,x=0,1,2,\ldots N\text{-}1$ **(I.138)**

Les premiers ordres n=0 et n=1 des polynômes normalisés discrets de Hahn sont calculés à partir de la relation précédente (I.138) comme suit :

$$\tilde{h}_0^{(a,b)}(x,N)=\sqrt{\frac{w(x)}{d_0^2}}\quad et\quad \tilde{h}_1^{(a,b)}(x,N)=[(a+b+2)x-(b+1)(N-1)]\sqrt{\frac{w(x)}{d_1^2}}$$ **(I.139)**

Les autres valeurs des polynômes de Hahn $2 \le n \le N-1$ sont déduites de la relation de récurrence par rapport à l'ordre n définie par l'équation (I.130).

La figure (I.14) représente les courbes des six premiers ordres des polynômes orthonormés de Hahn avec différentes valeurs des paramètres $a \; et \; b$.

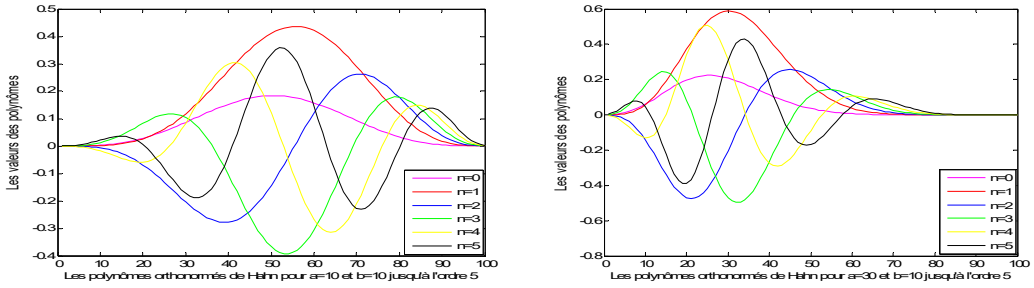


Figure (I.14): Courbes des polynômes orthonormés de Hahn pour ($a = b = 10$) et ($a = 30$; $b = 10$) pour les six premiers ordres.

3.1.9. Les polynômes de Meixner

Les polynômes orthogonaux discrets de Meixner $M_n^{(\beta,\mu)}(x)$ d'une seule variable satisfont à l'équation de différences partielles du second ordre suivante [Nikiforov et al., 1991; Zhu et al., 2010]:

$$x\Delta\nabla M_n^{(\beta,\mu)}(x) + (\beta\mu - x(1-\mu))\Delta M_n^{(\beta,\mu)}(x) + n(1-\beta)M_n^{(\beta,\mu)}(x) = 0$$
$$avec \qquad x, n = 0,1,2\ldots..N \quad , \quad \beta > 0 \quad et \quad 0 < \mu < 1 \qquad \textbf{(I.140)}$$

Le nième polynôme de Meixner est défini en utilisant la fonction hypergeometrique ${}_2F_1$ par la relation suivante :

$$M_n^{(\beta,\mu)}(x) = (\beta)_n \, {}_2F_1(-n, -x; \beta; 1 - 1/\mu) \qquad \textbf{(I.141)}$$

Les premiers ordres n=0 et n=1 des polynômes normalisés discrets de Meixner sont calculés à partir de l'équation (I.141) par :

$$\tilde{M}_0^{(\beta,\mu)}(x) = \sqrt{\frac{w(x)}{d_0^2}} \quad \text{et} \quad \tilde{M}_1^{(\beta,\mu)}(x) = (\beta + x - \frac{x}{\mu})\sqrt{\frac{w(x)}{d_1^2}} \qquad \textbf{(I.142)}$$

Les autres valeurs des polynômes de Meixner $n \geq 2$ sont déduites de la relation de récurrence par rapport à l'ordre n définie par l'équation (I.130).

La figure (I.15) représente les courbes des six premiers ordres des polynômes orthonormés de Meixner avec différentes valeurs des paramètres β et μ.

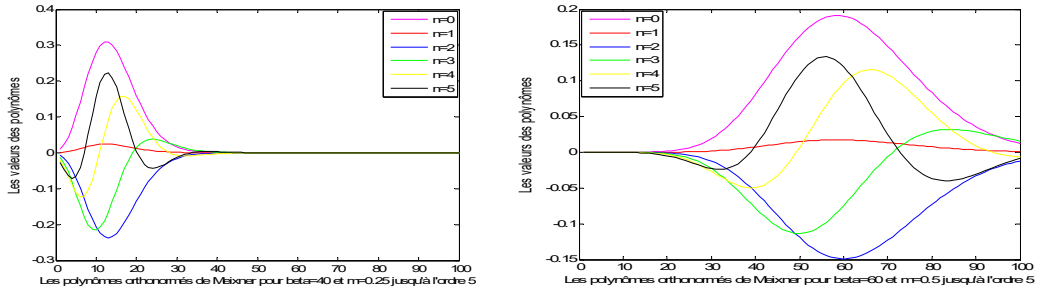


Figure (I.15): Courbes des polynômes orthonormés de Meixner pour $\mu = 0.25, \beta = 40$ et $\mu = 0.5, \beta = 60$ pour les six premiers ordres.

3.1.10. Les polynômes de Charlier

Les polynômes orthogonaux discrets de Charlier $C_n^{a_1}(x)$ d'une seule variable satisfont à l'équation de différences partielles du second ordre suivante [Nikiforov et al., 1991; Zhu et al., 2010]:

$$x\Delta\nabla C_n^{a_1}(x) + (a_1 - x)\Delta C_n^{a_1}(x) + nC_n^{a_1}(x) = 0$$
$$\text{avec} \qquad x, n = 0,1,2\ldots..N \quad , \quad a_1 > 0 \qquad \textbf{(I.143)}$$

Le nième polynôme de Charlier $C_n^{a_1}(x)$ est défini en utilisant la fonction hypergéométrique ${}_2F_0$ par la relation suivante :

$$C_n^{a_1}(x) = {}_2F_0(-n, -x; ; 1/a_1) \qquad \textbf{(I.144)}$$

Les premiers ordres n=0 et n=1 des polynômes normalisés discrets de Charlier $C_n^{a_1}(x)$ sont calculés à partir de l'équation précédente comme suit :

$$\tilde{C}_0^{a_1}(x) = \sqrt{\frac{w(x)}{d_0^2}} \quad \text{et} \quad \tilde{C}_1^{a_1}(x) = \frac{a_1 - x}{a_1}\sqrt{\frac{w(x)}{d_1^2}} \tag{I.145}$$

Les autres valeurs des polynômes de Charlier $n \geq 2$ sont déduites de la relation de récurrence par rapport à l'ordre n définie par l'équation (I.130).

La figure (I.16) représente les courbes des six premiers ordres des polynômes orthonormés de Charlier avec deux valeurs différentes du paramètre a.

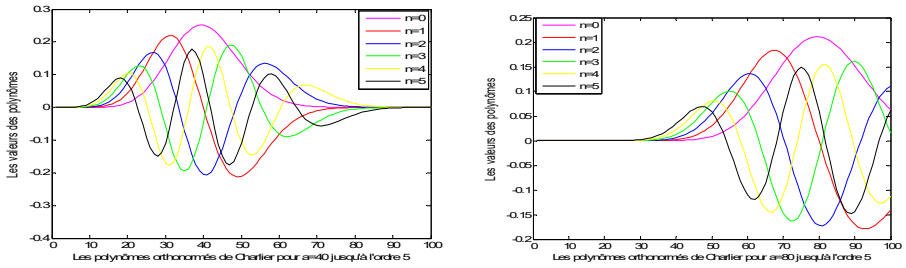


Figure (I.16): Courbes des polynômes orthonormés de Charlier pour $a_1 = 40$ et $a_1 = 80$ pour les six premiers ordres.

3.1.1. Relation de récurrence par rapport à la variable x

Le calcul des valeurs des polynômes orthogonaux d'une variable discrète en utilisant la relation de récurrence par rapport à l'ordre n pour les grands ordres nécessite beaucoup de temps de calcul et provoque la propagation des erreurs numériques. Pour remédier à ce problème, Mukundan a proposé une solution dans [Mukundan et al., 2001] pour calculer les polynômes de Tchebichef en utilisant la relation de récurrence par rapport à la variable x au lieu de la relation de récurrence par rapport à l'ordre n. Zhu a proposé également la même stratégie pour accélérer le calcul des polynômes de Krawtchouk, Hahn, Charlier & Meixner [Zhu et al., 2010].

Dans cette section, on va présenter la forme générale de la relation de récurrence par rapport à la variable x pour calculer les polynômes orthogonaux d'une variable discrète de Tchebichef, krawtchouk, Hahn, Meixner et Charlier.

En considérant les propriétés des opérateurs de différence en arrière ∇ et de différence en avance Δ, nous avons la relation suivante :

$$\Delta\nabla P_n(x) = P_n(x+1) - 2P_n(x) + P_n(x-1) \tag{I.146}$$

Donc, la relation de récurrence des polynômes orthogonaux discrets par rapport à la variable x est obtenue à partir des équations (I.23) et (I.146) comme suit:

$$p_n(x;N) = \frac{2\sigma(x-1)+\tau(x-1)-\lambda_n}{\sigma(x-1)+\tau(x-1)}\, p_n(x-1;N) - \frac{\sigma(x-1)}{\sigma(x-1)+\tau(x-1)}\, p_n(x-2;N) \tag{I.147}$$

On se basant sur les équations (I.130) et (I.147), on peut obtenir la forme générale de la relation de récurrence par rapport à la variable x des polynômes orthonormés discrets :

$$\tilde{p}_n(x;N) = \frac{\sqrt{w(x)}}{\sigma(x-1)+\tau(x-1)} \times \left[\frac{2\sigma(x-1)+\tau(x-1)-\lambda_n}{\sqrt{w(x-1)}}\, \tilde{p}_n(x-1;N) - \frac{\sigma(x-1)+\tau(x-1)}{\sqrt{w(x-2)}}\, \tilde{p}_n(x-2;N) \right] \tag{I.148}$$

Par suite, les relations de récurrence des polynômes orthonormés discrets de Tchebichef, Krawtchouk, Hahn, Meixner et Charlier par rapport à la variable x est obtenue à partir de la formule générale (I.148) et les données du tableau (I.1).

En utilisant la formule de Rodriguez (I.123), la relation d'orthogonalité (I.131) et le Tableau (I.1), on obtient les valeurs initiales de relation de récurrence par rapport à la variable x énumérées dans le tableau (I.3). Les équations ci-dessus peuvent être utilisées pour calculer les polynômes orthonormés discrets de Tchebichef pondérés, Krawtchouk, Hahn, Meixner et Charlier.

Tableau (I.3) : Les valeurs initiales de la relation de récurrence à trois termes par rapport à la variable x pour les polynômes de Tchebichef, Krawtchouk, Hahn, Meixner et Charlier.

$\tilde{P}_n(x)$	$x=0$	$x=1$
$\tilde{t}_n(\mathrm{x};\mathrm{N})$	$(-1)^n (N-n)_n \sqrt{\frac{w(0)}{d_n^2}}$	$(1+\frac{n(n+1)}{1-N})\sqrt{\frac{w(1)}{w(0)}}\tilde{t}_n(0;N)$
$\tilde{K}_n(\mathrm{x};\mathrm{p},\mathrm{N})$	$(-1)^n \binom{N}{n} p^n \sqrt{\frac{w(0)}{\rho(n,N)}}$	$\frac{(N-p)p-n(1-p)}{Np}\sqrt{\frac{w(1)}{w(0)}}\tilde{K}_n(0;p,N)$
$\tilde{C}_n^{a_1}(x)$	$\sqrt{\frac{w(0)}{d_n^2}}$	$\frac{a_1-n}{a_1}\sqrt{\frac{w(1)}{w(0)}}\tilde{C}_n^{a_1}(0)$
$\tilde{M}_n^{(\beta,\mu)}(x)$	$(\beta)_n\sqrt{\frac{w(0)}{d_n^2}}$	$\frac{\mu(n+\beta)-n}{\mu\beta}\sqrt{\frac{w(1)}{w(0)}}M_n^{(\beta,\mu)}(0)$
$\tilde{h}_n^{(a,b)}(x,N)$	$(1-N)_n\binom{n+b}{n}\sqrt{\frac{w(0)}{d_n^2}}$	$\frac{(n+b+1)(N-n-1)-n(N+a-1)}{(b+1)(N-1)}\sqrt{\frac{w(1)}{w(0)}}\tilde{h}_n^{(a,b)}(0,N)$

3.2. Les moments orthogonaux discrets

Dans cette section, nous présentons une autre série des moments orthogonaux c'est les moments orthogonaux discrets basés sur des polynômes orthogonaux classiques d'une variable discrète comme les polynômes de Tchebichef, Krawtchouk, Racah, Dual Hahn, Charlier, Mexnier, et Hahn. L'usage des polynômes discrets orthogonaux comme fonctions de base pour le calcul des moments de l'image, élimine le besoin de l'approximation numérique des intégrales par des sommations et satisfait exactement la propriété de l'orthogonalité dans le domaine discret de l'espace des coordonnées de l'image. Cette propriété rend les moments orthogonaux discrets supérieurs aux moments non orthogonaux (géométriques et complexes) et moments orthogonaux continus conventionnels dans le sens de la capacité de la représentation de l'image.

Les moments orthogonaux discrets unidimensionnels (1-D) sont définis par :

$$M_n = \sum_{x=0}^{N-1} \tilde{P}_n(x) f(x), \quad n = 0,1,...,s \tag{I.149}$$

où $f(x)$ est un signal de dimension 1 (1-D) et de longueur N.

La reconstruction d'un signal 1-D à partir d'un ensemble des moments orthogonaux discrets M_n d'ordre max est donnée par :

$$\hat{f}(x) = \sum_{n=0}^{\max} M_n \tilde{P}_n(x), \quad n = 0,1,...,N-1 \tag{I.140}$$

Les moments orthogonaux discrets MD_{nm} de deux dimensions (2-D) d'ordre $(n+m)$ d'une image d'intensité $f(x,y)$ de taille $M \times N$ sont donnés par la relation suivante:

$$MD_{nm} = \frac{1}{\rho(n,M)\rho(m,N)} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} Poly_n(x) Poly_m(y) f(x,y) \tag{I.141}$$

où $P_n(x)$ est Le nième polynôme orthogonal d'une variable discrète x qui peut être de Tchebichef, Charlier, Mexnier, Krawtchouk, Racah et Hahn et $\rho(n,M)$ est le facteur de normalisation de polynôme $P_n(x)$.

Selon le type de polynômes utilisé en (I.141), on obtient des moments orthogonaux discrets tels que les moments de : Tchebichef, Charlier, Mexnier, Krawtchouk, Racah, et Hahn.

Les sections suivantes décrivent brièvement les moments orthogonaux discrets de Tchebichef, Charlier, Mexnier, Krawtchouk, Racah, et Hahn. Ces moments sont définis à partir des fonctions hypergéométriques généralisées.

3.2.1. Les moments de Tchebichef

Mukundan et al. [Mukundan et al., 2001] sont les premiers qu'ont proposés un ensemble de fonctions des moments orthogonaux discrets basé sur les polynômes orthogonaux discrets de Tchebitchef, pour les tâches de l'analyse de l'image. Dans les années qui suivent, les moments de Tchebichef ont été appliqués par plusieurs chercheurs pour la reconstruction d'image [Hmimid et al, 2012; Mukundan, 2004], classification des objets [Zhu et al., 2007b ;] et le tatouage d'image [Deng et al., 2009]. Les polynômes orthogonaux Tchebichef sont définis dans un domaine discret en termes des fonctions hypergéométriques par la formule suivante :

$$t_n(x) = (1-N)_n\, {}_3F_2(-n,-x,1+n;1,1-N;1) = \sum_{k=0}^{n} (-1)^{n-k} \binom{N-1-k}{n-k} \binom{n+k}{n} \binom{x}{k} \quad \textbf{(I.142)}$$

où ${}_3F_2$ est la fonction hypergéométrique généralisée, $n, x = 0,1,2,...,N-1$, et N est la taille de l'image.

Pour garder la stabilité numérique dans une gamme dynamique limitée, on va utiliser les polynômes de Tchebichef normalisées qui sont définis par la relation suivante :

$$\tilde{t}_n(x) = \frac{\tilde{t}_n(x)}{\beta(n,N)} \quad \textbf{(I.143)}$$

avec $\beta(n,N)$ est une constante appropriée indépendante de x qui sert comme facteur d'échelle, tel que $\beta(n,N)=N^n$.

Le facteur de normalisation $\tilde{\rho}(n,M)$ des polynômes de Tchebichef est défini par la relation suivante :

$$\tilde{\rho}(n,M)=\frac{\rho(n,M)}{\beta(n,N)} \quad ; \quad n=0,1,2,...,N-1 \tag{I.144}$$

où $\rho(n,M)=(2n)!\binom{N+n}{2n+1}$

Selon la formule générale des moments orthogonaux discrets, donnée par l'équation (I.141), les moments Tchebichef normalisées d'ordre $(n+m)$ sont définis par la relation suivante :

$$MT_{nm}=\frac{1}{\tilde{\rho}(n,M)\tilde{\rho}(m,N)}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\tilde{t}_n(x)\tilde{t}_m(y)f(x,y) \tag{I.145}$$

3.2.2. Les moments de Krawtchouk

Les moments orthogonaux discrets de Krawtchouk (KM) sont le second type des moments discrets introduits dans l'analyse d'image par Yap et al. [Yap et al., 2003] . Après, ils sont largement utilisés par d'autres chercheurs pour le reconstruction d'image [Zhu et al., 2010 ; Sayyouri et al., 2012d] et la classification des images [Papakostas et al., 2010]. Ces moments se basent sur les polynômes orthogonaux discrets de Krawtchouk définis à partir des fonctions hypergéométriques par la relation suivante :

$$k_n(x;p,N)={}_2F_1\left(-n,-x,-N;\frac{1}{p}\right) \tag{I.146}$$

où $x,n=0,1,2,...,N, N>0, p\in(0,1)$.

De même pour les polynômes de Tchebichef, le calcul des polynômes de Krawtchouk à l'aide de l'équation (I.141) présente des fluctuations numériques. Donc, on va

utiliser une version plus appropriée de ces polynômes, c'est les polynômes pondérés de Krawtchouk définis par :

$$\tilde{K}_n(x;p,N) = K_n(x;p,N)\sqrt{\frac{w(x;p,N)}{\rho(x;p,N)}} \tag{I.147}$$

où $w(x;p,N)$ la fonction de poids définie par :

$$w(x;p,N) = \binom{N}{x} p^x (1-p)^{N-x} \tag{I.148}$$

et $\rho(n;p,N)$ est le facteur de normalisation défini par :

$$\rho(n;p,N) = (-1)^n \left(\frac{1-p}{p}\right)^n \frac{n!}{(-N)_n} \tag{I.149}$$

Selon la formule générale des moments orthogonaux discrets donnée par l'équation (I.141), les moments de krawtchouk de l'ordre $(n+m)$ en termes de polynômes pondérés de krawtchouk, pour une image d'intensité $f(x,y)$ et de taille $M \times N$ sont définies par :

$$MK_{nm} = \sum_{x=0}^{N-1}\sum_{y=0}^{M-1} \tilde{k}_n(x;p_1,N-1)\,\tilde{k}_m(y;p_2,M-1)\,f(x,y). \tag{I.150}$$

3.2.3. Les moments de Racah

Récemment, Zhu et al. [Zhu et al., 2007a] ont proposé une nouvelle classe des moments orthogonaux discrets pour les problèmes de traitement d'image, qui utilise les polynômes orthogonaux de Racah comme des fonctions de base. Ces moments sont appelés les moments orthogonaux discrets de Racah.

Le polynôme de Racah $u_n^{(\alpha,\beta)}(s,a,b)$ d'ordre n, est défini explicitement comme une somme hypergéométrique généralisée [Aquilanti et al., 2001]:

$$u_n^{(\alpha,\beta)}(s,a,b)=\frac{1}{n!}(a-b+1)_n(\beta+1)_n(a+b+\alpha+1)_n \times_4 F_3\left(\begin{matrix}-n,\alpha+\beta+n+1,a-s,a+s+1\\ \beta+1,a+1-b,a+b+\alpha+1\end{matrix};1\right)$$

n=0, 1, …, L-1; s=a, a+1, …, b-1; **(I.151)**

Les polynômes de Racah satisfont la propriété de l'orthogonalité suivante :

$$\sum_{s=a}^{b-1} u_n^{(\alpha,\beta)}(s,a,b)u_m^{(\alpha,\beta)}(s,a,b)\rho(s)\left[\Delta x(s-\frac{1}{2})\right]=\delta_{nm}d_n^2 \quad \textbf{(I.152)}$$

où

$$d_n^2=\frac{\Gamma(\alpha+n+1)\Gamma(\beta+n+1)\Gamma(b-a+\alpha+\beta+n+1)\Gamma(a+b+\alpha+n+1)}{(\alpha+\beta+2n+1)n!(b-a-n-1)!\Gamma(\alpha+\beta+n+1)\Gamma(a+b-\beta-n)} \quad \text{n=0,1,..,L-1}$$

(I.153)

L'ensemble des polynômes de Racah n'est pas approprié à définir des moments parce que la gamme des valeurs des polynômes augmente rapidement avec l'augmentation de l'ordre. Une méthode habituelle pour éviter des fluctuations numériques pour les calculs de moment est la normalisation.

Pour éviter l'instabilité numérique dans le calcul polynômial, les polynômes de Racah sont normalisés en utilisant la norme carrée et la fonction de pondération. L'ensemble des polynômes normalisés de Racah sont définis par :

$$\hat{u}_n^{(\alpha,\beta)}(s,a,b)=u_n^{(\alpha,\beta)}(s,a,b)\sqrt{\frac{\rho(s)}{d_n^2}\left[\Delta x(s-\frac{1}{2})\right]} \quad \text{n=0, 1, …, L-1,} \quad \textbf{(I.154)}$$

Dans ce cas, la condition de l'orthogonalité donnée par l'équation (I.32) devient

$$\sum_{s=a}^{b-1}\hat{u}_n^{(\alpha,\beta)}(s,a,b)\hat{u}_m^{(\alpha,\beta)}(s,a,b)=\delta_{nm} \quad \text{n,m = 0,1,…, L-1} \quad \textbf{(I.155)}$$

où s une variable uniforme désignant le pas s=a, a+1, …, b–1. Les valeurs des polynômes normalisés de Racah sont ainsi définies dans l'intervalle [- 1, 1].

Les moments de Racah U_{nm} d'ordre $(n+m)$ d'une image d'intensité $f(s,t)$ et de dimension $M \times N$ sont donnés par :

$$U_{nm} = \sum_{s=a}^{b-1} \sum_{t=a}^{b-1} \hat{u}_n^{(\alpha,\beta)}(s,a,b) \hat{u}_m^{(\alpha,\beta)}(t,a,b) f(s,t) \quad \text{n,m} = 0,1,\ldots, \text{L-1}, \tag{I.156}$$

3.2.4. Les moments de Charlier

Les moments orthogonaux discrets de Charlier (CM) sont définis à partir des polynômes orthogonaux discrets de Charlier et sont utilisés pour les taches de l'analyse d'image par Zhu et al. [Zhu et al., 2010] et Sayyouri et al., [Sayyouri et al., 2012b ; 2012c]. Les polynômes orthogonaux discrets de Charlier sont définis à partir des fonctions hypergéométriques par la relation suivante :

$$C_n^{a_1}(x) = {}_2F_0(-n,-x;;1/a_1) \tag{I.157}$$

où $x, n = 0,1,2,\ldots,N, \quad a_1 \succ 0$

De même manière que les polynômes Krawtchouk, le calcul des polynômes de Charlier à l'aide de l'équation (I.141) présente des fluctuations numériques. Pour limiter cette fluctuation, on va utiliser les polynômes normalisés de Charlier définis par :

$$\tilde{C}_n^{a_1}(x) = C_n^{a_1}(x) \sqrt{\frac{w(x)}{d_n^2}} \tag{I.158}$$

où $w(x;p,N)$ la fonction de poids définie par :

$$w(x) = \frac{e^{-a_1} a_1^x}{x!} \tag{I.159}$$

et d_n^2 est le facteur de normalisation défini par :

$$d_n^2 = \frac{n!}{a_1^n} \tag{I.160}$$

Selon la formule générale des moments orthogonaux discrets donnée par l'équation (I.141), les moments de Charlier de l'ordre $(n+m)$ en termes de polynômes normalisés de Charlier, pour une image d'intensité $f(x,y)$ et de taille $M\times N$ sont définies par :

$$MC_{nm}=\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\tilde{C}_n^{a_1}(x)\tilde{C}_m^{a_1}(y)f(x,y) \tag{I.161}$$

3.2.5. Les moments de Hahn

Dans cette sous-section on va présenter une autre série des moments orthogonaux discrets, c'est les moments de Hahn (MH) qui sont introduits dans l'analyse d'image pour les taches de reconstruction et de classification [Yap et al, 2007, Sayyouri et al., 2012a, Sayyouri et al., 2013a,]. Ces moments utilisent les polynômes orthogonaux discrets de Hahn définis à partir des fonctions hypergéométriques par la relation suivante [Nikiforov et al., 1991]:

$$h_n^{(a,b)}(x,N)=\frac{(1)^n(b+1)_n(N-n)_n}{n!}\,{}_3F_2(-n,-x,n+1+a+b;b+1,1-N;1) \tag{I.162}$$

où $x,n=0,1,2,\ldots,N-1,\quad a\succ 0\quad \text{and}\quad b\succ 0$

Pour avoir la stabilité numérique des valeurs des polynômes de Hahn, on va utiliser les polynômes normalisés de Hahn définis par :

$$\tilde{h}_n^{(a,b)}(x,N)=h_n^{(a,b)}(x,N)\sqrt{\frac{w(x)}{d_n^2}} \tag{I.163}$$

où $w(x;p,N)$ la fonction de poids définie par :

$$w(x)=\frac{\Gamma(N+a-x)\Gamma(b+1+x)}{\Gamma(N-x)\Gamma(x+1)} \tag{I.164}$$

et d_n^2 est le facteur de normalisation défini par

$$d_n^2=\frac{\Gamma(a+n+1)\Gamma(b+n+1)(a+b+n+1)_N}{(a+b+2n+1)n!(N-n-1)!} \tag{I.165}$$

Les moments de Hahn de l'ordre $(n+m)$ en termes de polynômes normalisés de Hahn, pour une image d'intensité $f(x,y)$ et de taille $M \times N$ sont définies par :

$$MH_{nm} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \tilde{h}_n^{(a,b)}(x,M) \tilde{h}_m^{(a,b)}(y,N) f(x,y) \quad \textbf{(I.166)}$$

Yap et al En 2007 ont introduit les moments orthogonaux de Hahn basés sur les polynômes orthogonaux discrets de Hahn pondérés [Yap et al, 2007]. Une méthode efficace pour le calcul des moments discrets de Hahn a été également développée par Sayyouri et al., en 2012 dans [Sayyouri et al., 2012a]. Notons que les moments orthogonaux discrets de Tchebichef et Krawtchouk sont des cas particuliers des moments de Hahn avec des paramètres appropriés.

4. Conclusion

Dans ce chapitre, nous avons exploré l'état d'art de l'existant en ce qui concerne les moments orthogonaux continues et discrets en partant des bases orthogonales jusque là étudiés.

Dans le reste de cet ouvrage, nous présentons l'évolution de nos recherches, pour l'extraction des moments de Meixner sur des des images 2D et 3D, et leur reconstruction ainsi que leurs classfications.

CHAPITRE 2: CALCUL RAPIDE DES MOMENTS DE MEIXNER ET LEURS INVERSES POUR DES IMAGES 2D

Les moments orthogonaux sont reconnus comme des outils utiles pour la représentation d'objets et l'analyse d'images. Il a été démontré qu'ils ont une meilleure capacité de représentation d'image que les moments orthogonaux continus. Un problème concernant l'utilisation des moments est le coût de calcul élevé, qui peut limiter les cas où le calcul en ligne est requis. Dans ce chapitre, nous proposons une méthode récursive basée sur la formule de récurrence de Clenshaw qui peut être implémentée pour transformer les noyaux des moments de Meixner et son inverse pour un calcul rapide. La méthode proposée ne nécessite pas de calculer les valeurs polynomiales de Meixner de divers ordres sur divers points de données, où la complexité de calcul est réduite. Les résultats expérimentaux montrent que la méthode proposée donne de meilleurs résultats que les méthodes existantes en termes de rapidité de calcul et d'efficacité de la capacité de reconstruction d'image dans des conditions sans bruit ou avec bruit.

1. Introduction

Le calcul des moments orthogonaux discrets est limité par deux difficultés majeures. La première est liée au coût de calcul élevé, en particulier pour les images de grande taille et aux ordres plus élevées. Ainsi, le calcul des moments prend beaucoup de temps et la complexité de calcul augmente en augmentant l'ordre des moments. La seconde est liée à la propagation d'erreur numérique dans le calcul de polynômes orthogonaux discrets. En effet, les valeurs calculées des polynômes orthogonaux discrets, utilisant la fonction hypergéométrique et la relation de récurrence de trois termes par rapport à l'ordre n des polynômes orthogonaux discrets, sont complexes et nécessitent un temps de calcul plus long pour tous les ordres supérieurs, ce qui entraîne la propagation des erreurs numériques [Zhu, 2010]. Pour limiter cette erreur et réduire le temps de calcul des moments, ainsi que pour améliorer la précision de la reconstruction d'image, les scientifiques proposent d'utiliser la relation de récurrence par rapport à la variable x au lieu de l'ordre n dans le calcul de polynômes orthogonaux discrets [Mukundan, 2004 ; Zhu, 2010].

Pour réduire le coût en temps de calcul des moments, plusieurs algorithmes sont introduits dans la littérature [Aburdene, 1995 ;Chau, 1994 ;Chau, 2000 ;Nikolajevic, 2003 ;Wang, 2004]. Wang et al. [Wang, 1994] a proposé des algorithmes récursifs pour le calcul des transformées en cosinus discrètes (DCT). Vladimir Nikolajevic et al. a proposé un autre algorithme pour le calcul de la transformée en cosinus discrète modifiée directe et inverse à l'aide de la formule de Clenshaw. Un algorithme de calcul récursif parallèle de la transformation inverse de moments de Legendre pour la reconstruction du signal et de l'image en utilisant la formule de récurrence de Clenshaw a été proposé dans son papier [Wang, 2004]. Récemment, Guobao Wang et ses collaborateurs [Wang, 2006] ont proposé un algorithme récursif pour le calcul des moments de Tchebichef et de sa transformée inverse à l'aide de la formule de Clenshaw. Cependant, aucun n'a exploré cette méthode pour le calcul rapide des moments de Meixner et de sa transformée inverse.

Dans ce chapitre, nous allons présenter un algorithme récursif pour le calcul rapide des moments de Meixner et sa transformée inverse pour la reconstruction d'image en utilisant la formule de récurrence de Clenshaw. Ainsi, l'application de cet algorithme nous a permis d'accélérer considérablement le temps de calcul des moments de Meixner par rapport à la méthode direct, et par rapport à la méthode récursif par x, tout en réduisant la complexité. Ainsi, les résultats des simulations montrent que l'approche proposée est très efficace en termes de temps de calcul des moments de Meixner par rapport aux méthodes existantes.

2. Les polynômes et moments de Meixner

Dans cette section, nous tenons à rappeler la définition des polynômes de Meixner, les moments de Meixners et leurs inverses pour les images 2D.

2.1. Les polynômes de Meixner

Le nième polynôme orthogonal discret de Meixner $M_n^{(\gamma,\mu)}(x)$ satisfait l'équation des différences partielles de premier-ordre suivante [Nkiforov, 1991]:

$$\sigma(x)\Delta\nabla M_n^{(\gamma,\mu)}(x)+\tau(x)\Delta M_n^{(\gamma,\mu)}(x)+\lambda_n M_n^{(\gamma,\mu)}(x)=0 \qquad \textbf{(II.1)}$$

Les fonctions $\sigma(x)\ \tau(x)$ et λ_n sont définies par

$$\sigma(x)=x,\ \tau(x)=\gamma\mu-x(1-\mu),\quad \textbf{et}\quad \lambda_n=n(1-\mu) \tag{II.2}$$

Les polynômes de Meixner d'ordre n sont définis par une fonction hypergéométrique:

$$M_n^{(\gamma,\mu)}(x)=(\gamma)_n\ {}_2F_1(-n,-x;\gamma;1-1/\mu)=\sum_{k=0}^{n}\alpha_{k,n}^{(\gamma,\mu)}x^k \tag{II.3}$$

où $x,n=0,1,2,...,N-1,\quad \gamma>0,\quad 0\prec\mu\prec 1$ et la fonction ${}_2F_1$ est la fonction hypergéométrique généralisée qui est définie par :

$${}_2F_1(a,b;c;z)=\sum_{k=0}^{\infty}\frac{(a)_k(b)_k}{(c)_k}\frac{z^k}{k!} \tag{II.4}$$

Le symbole $(a)_k$ est le symbole de Pochhammer donné par :

$$(a)_k=a(a+1)(a+2)....(a+k-1)=\frac{\Gamma(a+k)}{\Gamma(a)} \tag{II.5}$$

Les polynômes orthogonaux discrets normalisés de Meixner $M_n^{(\gamma,\mu)}(x)$ satisfont la relation d'orthogonalité de la forme :

$$\sum_{x=0}^{N-1}\omega(x)M_n^{(\gamma,\mu)}(x)M_m^{(\gamma,\mu)}(x)=\rho(n)\delta_{nm}\qquad 0\prec n,m\prec N-1 \tag{II.6}$$

δ_{nm} Est la fonction de Dirac.

$\rho(n)$ Pour la norme carré du polynôme orthogonaux discrets normalisés de Meixner défini par :

$$\rho(n)=\frac{n!(\gamma)_n}{\mu^n(1-\mu)^{\gamma}} \tag{II.7}$$

$\omega(x)$ est la fonction de poids des polynômes discrets de Meixner

$$\omega(x) = \frac{\mu^x \Gamma(\gamma + x)}{\Gamma(1+x)\Gamma(\gamma)} \tag{II.8}$$

Les polynômes orthogonaux discrets normalisés de Meixner peuvent être obtenus par :

$$\tilde{M}_n^{(\gamma,\mu)}(x) = M_n^{(\gamma,\mu)}(x)\sqrt{\frac{\omega(x)}{\rho(n)}} \qquad n = 0,1,...,N-1 \tag{II.9}$$

2.1.1. Relation de récurrence par n

Puisque le calcul des polynômes de Meixner normalisés par l'équation (II.3) a un coût en temps de calcul élevé. Donc, pour calculer les valeurs des polynômes orthogonaux discrets de Meixner, nous utiliserons les relations de récurrence de trois termes par rapport à l'ordre n. En effet, les polynômes discrets de Meixner satisfont les relations de récurrence à trois termes suivantes par respect à l'ordre n [Sayyouri, 2012c]:

$$\tilde{M}_n^{(\gamma,\mu)}(x) = \frac{BD}{A}\tilde{M}_{n-1}^{(\gamma,\mu)}(x) - \frac{CE}{A}\tilde{M}_{n-2}^{(\gamma,\mu)}(x) \tag{II.10}$$

Les valeurs initiales sont définies par:

$$\begin{aligned} \tilde{M}_0^{(\gamma,\mu)}(x) &= \sqrt{\frac{\omega(x)}{\rho(0)}} \\ \tilde{M}_1^{(\gamma,\mu)}(x) &= \left(\gamma + x - \frac{x}{\mu}\right)\sqrt{\frac{\omega(x)}{\rho(1)}} \end{aligned} \tag{II.11}$$

Les coefficients A, B, C, D et E sont définies comme suit:

$$A = \frac{\mu}{\mu - 1};$$

$$B = \frac{x(1-\mu) - \mu(n+\gamma-1) - n + 1}{1-\mu};$$

$$C = \frac{(n-1)(n-2+\gamma)}{\mu - 1};$$

$$E=\sqrt{\frac{\mu^2}{n(n-1)(n+\gamma-1)(n+\gamma-2)}}$$

$$D=\sqrt{\frac{\mu}{n(n+\gamma-1)}}; \qquad \textbf{(II.12)}$$

2.1.2. Relation de récurrence par x

Les valeurs calculées des polynômes orthogonaux discrets de Meixner à l'aide des fonctions hypergéométriques et des relations de récurrence de trois termes par rapport à l'ordre n sont complexes et nécessitent un temps de calcul élevé, en particulier pour les ordres supérieurs, ce qui entraîne la propagation d'erreurs numériques [Zhu, 2010]. Pour limiter cette erreur, nous avons appliqué la relation de récurrence par rapport à variable x au lieu d'ordre n dans le calcul des polynômes orthogonaux discrets de Meixner. Ainsi, en considérant les propriétés des opérateurs Δ et ∇ , nous avons :

$$\Delta\nabla M_n^{(\gamma,\mu)}(x)=M_n^{(\gamma,\mu)}(x+1)-2M_n^{(\gamma,\mu)}(x)+M_n^{(\gamma,\mu)}(x-1) \qquad \textbf{(II.13)}$$

Ainsi, les relations de récurrence des polynômes orthogonaux discrets de Meixner par rapport à x peuvent être obtenues par les équations (II.1) et (II.13) comme suit:

$$M_n^{(\gamma,\mu)}(x)=\frac{(x-1)(1+\mu)+\gamma\mu-n(1-\mu)}{\mu(x+\gamma-1)}M_n^{(\gamma,\mu)}(x-1)-\frac{x-1}{\mu(x+\gamma-1)}M_n^{(\gamma,\mu)}(x-2) \qquad \textbf{(II.14)}$$

Les valeurs initiales de la relation de récurrence par rapport à la variable x sont définies comme suit:

$$M_n^{(\gamma,\mu)}(0)=(\gamma)_n$$
$$M_n^{(\gamma,\mu)}(1)=\left(1+\frac{n}{\gamma}\left(1-\frac{1}{\mu}\right)\right)M_n^{(\gamma,\mu)}(0) \qquad \textbf{(II.15)}$$

2.2. Les moments de Meixner

Le moment orthogonal discret unidimensionnel *1D* de Meixner *MM* de l'order *n* du signal discret $f(x)$ avec la dimension *N* est défini par :

$$MM_n = \sum_{x=0}^{N-1} \tilde{M}_n^{(\gamma,\mu)}(x) f(x) \tag{II.16}$$

Le moment de Meixner d'ordre $(n+m)$ en terme du poids du polynôme de Meixner, pour une image associé à une fonction d'intensité $f(x,y)$ d'une dimension $M \times N$ est defini par :

$$MM_{nm} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \tilde{M}_n^{(\gamma,\mu)}(x) \tilde{M}_m^{(\gamma,\mu)}(y) f(x,y) \tag{II.17}$$

où $M_n^{(\gamma,\mu)}(x)$ est le polynôme orthogonal discret de Meixner d'ordre *n*.

2.3. La transformée inverse

Par la solution de l'équation (II.16) et l'équation d'orthogonalité, l'ensemble des moments de Meixner MM_n, pour le signal discret $f(x)$, la transformée inverse peut être représenté par :

$$f(x) = \sum_{n=0}^{N-1} MM_n \tilde{M}_n^{(\gamma,\mu)}(x) \tag{II.18}$$

La fonction d'intensité $f(x,y)$ d'une image 2D peut être représentée par la transformée inverse en terme des moments de Meixner comme suit :

$$f(x,y) = \sum_{n=0}^{M-1} \sum_{m=0}^{N-1} MM_{nm} \tilde{M}_n^{(\gamma,\mu)}(x) \tilde{M}_m^{(\gamma,\mu)}(y) \tag{II.19}$$

Une mesure objective est utilisée pour mesurer l'erreur de l'image reconstruite par rapport à l'image d'origine. Étant donné la fonction d'intensité $f(x,y)$ de l'image d'origine aux

coordonnées (x,y) et $\tilde{f}_{rec}(x,y)$ correspond à la fonction d'intensité de l'image reconstruite, le MSE est défini comme suit:

$$MSE = \frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left(\tilde{f}_{rec}(x,y) - f(x,y) \right)^2 \quad \textbf{(II.20)}$$

où $M \times N$ les dimensions de l'image.

L'évaluation de la transformée du moment de Meixner, si l'on utilise la relation de récurrence vis-à-vis de n et x, nécessite des opérations assez importantes pour le calcul des polynômes de Meixner de divers ordres sur divers points de données, ce qui ralentit l'évaluation. Pour accélérer le calcul des moments de Meixner, nous allons utiliser la formule de récurrence de Clenshaw, pour transformer le calcul des moments discrets de Meixner et sa transformée inverse, en proposant un algorithme de calcul considérablement rapide.

3. La formule de récurrence de Clenshaw

La formule de récurrence de Clenshaw [Wang, 2006] est une formule efficace pour évaluer la somme des produits des coefficients indexés par les fonctions qui obéit à la relation suivante:

$$Q(x) = \sum_{n=0}^{N-1} c_n F_n(x) \quad \textbf{(II.21)}$$

où $F_n(x)$ obéit à la relation de récurrence suivante:

$$F_{n+1}(x) = \alpha(n,x) F_n(x) + \beta(n,x) F_{n-1}(x) \quad \textbf{(II.22)}$$

Pour des fonctions $\alpha(n,x)$ et $\beta(n,x)$. Nous définissons les quantités ψ_n en suivant les récurrences:

$$\begin{aligned} &\psi_{N+1} = \psi_N = 0 \\ &\psi_n = \alpha(n,x)\psi_{n+1} + \beta(n+1,x)\psi_{n+2} + c_n \end{aligned} \quad \textbf{(II.23)}$$

pour $n = N-1,....,1$ et résoudre en arrière (backward) pour obtenir ψ_1 et ψ_2.

La formule de récurrence de Clenshaw à la baisse (downward) indique ensuite que la somme $Q(x)$ peut être calculée par :

$$Q(x) = \beta(1,x) F_0(x) \psi_2 + F_1(x) \psi_1 + c_0 F_0(x) \tag{II.24}$$

Dans la section suivante, nous présenterons une nouvelle formule permettant de calculer rapidement les moments orthogonaux discrets de Meixner à l'aide de la formule de récurrence de Clenshaw.

3.1. Calcul des moments de Meixner via la relation de récurrence de Clenshaw

Semblable au principe de dérivation de l'algorithme de signal 1D, nous pouvons obtenir l'algorithme pour le calcul de moments Meixner 2D. En raison de la séparabilité des polynômes de Meixner 2D, la relation récursive dans l'équation (II.14) est dérivée. Ensuite, en utilisant la définition des fonctions pondérées et normalisées dans l'équation (II.9), qui peut être réécrite comme une forme de récurrence à trois termes, nous obtenons l'expression suivante:

$$\tilde{M}_n^{(\gamma,\mu)}(x+1) = H \frac{x(1+\mu)+\gamma\mu-n(1-\mu)}{\mu(x+\gamma)} \tilde{M}_n^{(\gamma,\mu)}(x) - R \frac{x}{\mu(x+\gamma)} \tilde{M}_n^{(\gamma,\mu)}(x-1) \tag{II.25}$$

où

$$H = \sqrt{\frac{\mu(\gamma+x)}{(x+1)}}; \quad \text{and} \quad R = \sqrt{\frac{\mu^2(\gamma+x)(\gamma+x-1)}{x(x+1)}} \tag{II.26}$$

avec

$$\begin{aligned} \tilde{M}_n^{(\gamma,\mu)}(0) &= \frac{(\gamma)_n}{\sqrt{\rho(n)}} \\ \tilde{M}_n^{(\gamma,\mu)}(1) &= \left(1+\frac{n}{\gamma}\left(1-\frac{1}{\mu}\right)\right)\sqrt{\mu\gamma}\,\tilde{M}_n^{(\gamma,\mu)}(0) \end{aligned} \tag{II.27}$$

Pour le cas des polynômes de Meixner, soit $F_x(n) = \tilde{M}_n^{(\gamma,\mu)}(x)$. En comparant l'équation (II.22) avec l'équation (II.25), nous avons :

$$\alpha_a(x,n) = H\frac{x(1+\mu)+\gamma\mu-n(1-\mu)}{\mu(x+\gamma)} \quad \textbf{et} \quad \beta_a(x,n) = \beta_a(x) = -R\frac{x}{\mu(x+\gamma)} \tag{II.28}$$

Selon l'équation (II.23), nous pouvons définir pour $x = N-1,....,1$

$$\begin{aligned} &\psi_{N+1} = \psi_N = 0 \\ &\psi_x = \alpha_a(x,n)\psi_{x+1} + \beta_a(x+1)\psi_{x+2} + f(x) \end{aligned} \tag{II.29}$$

En utilisant la relation de récurrence défini dans (II.24) avec c_0 remplacé par $f(x)$, le moment de Meixner unidimensionnel 1D donné dans (I.16) peut être exprimé comme suit :

$$\begin{aligned} MM_n &= f(0)F_0(n) + F_1(n)\psi_1 + \beta_a(1)F_0(n)\psi_2 \\ &= f(0)\tilde{M}_n^{(\gamma,\mu)}(0) + \tilde{M}_n^{(\gamma,\mu)}(1)\psi_1 + \beta_a(1)\tilde{M}_n^{(\gamma,\mu)}(0)\psi_2 \end{aligned} \tag{II.30}$$

Notez que $\tilde{M}_n^{(\gamma,\mu)}(1) = \alpha_a(0,n)\tilde{M}_n^{(\gamma,\mu)}(0)$ où

$$\alpha_a(0,n) = \frac{\gamma\mu-n(1-\mu)}{\gamma\mu}\sqrt{\mu\gamma} \quad \text{et} \quad \beta_a(1) = -\sqrt{\frac{\mu^2(\gamma+1)\gamma}{2}}\frac{1}{\mu(1+\gamma)}$$

Selon l'équation (II.28), donc nous pouvons écrire :

$$MM_n = \tilde{M}_n^{(\gamma,\mu)}(0)\left[f(0) + \frac{\gamma\mu-n(1-\mu)}{\gamma\mu}\sqrt{\mu\gamma}\psi_1 - \sqrt{\frac{\mu^2(\gamma+1)\gamma}{2}}\frac{1}{\mu(1+\gamma)}\psi_2\right]. \tag{II.31}$$

Selon la formule de récurrence de Clenshaw (II.29) avec $x=0$, nous obtenons :

$$\begin{aligned} \psi_0 &= \alpha_a(0,n)\psi_1 + \beta_a(1)\psi_2 + f(0) \\ &= f(0) + \frac{\gamma\mu-n(1-\mu)}{\gamma\mu}\sqrt{\mu\gamma}\psi_1 - \sqrt{\frac{\mu^2(\gamma+1)\gamma}{2}}\frac{1}{\mu(1+\gamma)}\psi_2 \end{aligned} \tag{II.32}$$

donc, l'équation (31) peut être réécrite comme suit :

$$MM_n = \psi_0\tilde{M}_n^{(\gamma,\mu)}(0) \tag{II.33}$$

En termes de multiplications et additions, nous représentons la complexité analysé pour cette méthode par rapport à la méthode directe, dans le tableau (II.1)

Tableau (II.1) : Complexité de calcul des moment1D de Meixner MM_n de l'ordre n

Méthodes	**Multiplications**	**Additions**
Méthode directe	4N-4	2N-3
Formule de Clenshaw	2N+1	2N

Les moments de Meixner 2D donnée par l'équation (II.17) peuvent être exprimés par

$$MM_{nm} = \sum_{x=0}^{M-1} \tilde{M}_n^{(\gamma,\mu)}(x)\,\omega_x(m) \tag{II.34}$$

où

$$\omega_x(m) = \sum_{y=0}^{N-1} \tilde{M}_m^{(\gamma,\mu)}(y) f(x,y) \tag{II.35}$$

Ainsi les coefficients $\omega_x(m)$ définis dans l'équation (II.35) sont initialement évalués pour $(x = M-1,....,1,0),$ selon l'algorithme des moments orthogonaux discrets de Meixner unidimensionnel présenté ci-dessus

$$\begin{aligned}
&\psi_{N+1} = \psi_N = 0 \\
&\psi_y = \alpha_a(y,m)\psi_{y+1} + \beta_a(y+1)\psi_{y+2} + f(x,y) \\
&\qquad (y = N-1,....,1,0), \\
&\omega_x(m) = \psi_0 \tilde{M}_m^{(\gamma,\mu)}(0)
\end{aligned} \tag{II.36}$$

Ainsi, $\omega_x(m)$ sont utilisés pour évaluer l'algorithme des moments orthogonaux discrets de Meixner à deux dimensions définis dans l'équation (II.34)

$$\begin{aligned}
&\varphi_{M+1} = \varphi_M = 0 \\
&\varphi_x = \alpha_a(x,n)\varphi_{x+1} + \beta_a(x+1)\varphi_{x+2} + \omega_x(m) \\
&\qquad (x = M-1,....,1,0), \\
&MM_{nm} = \varphi_0 \tilde{M}_n^{(\gamma,\mu)}(0)
\end{aligned} \tag{II.37}$$

Pour calculer un moment de Meixner 2D d'ordre $(n+m)$. Dans l'équation (II.36), l'évaluation de $\omega_x(m)$ nécessite 2N+1 multiplications et 2N additions pour chaque valeur de x. Comme il existe M valeurs de x, un total de 2NM+M multiplications et 2NM additions sont requises. La deuxième étape donnée par l'équation (II.37) nécessite 2M+1 multiplications et 2M additions. Par conséquence, le calcul d'un moment de Meixner MM_{nm} utilisant l'implémentation récursive proposée dans (II.36) et (II.37), pour un total de 2NM+3M+1 multiplications et 2NM+2M additions.

La méthode directe pour calculer un moment Meixner 2D est le calcul en deux temps, comme indiqué aux équations (II.34) et (II.35), ce qui donne un total de (4N-4)M+4M-4=4NM-4 multiplications et (2N-3)M+2M-3=2NM-M-3 additions. La même chose est donnée dans le tableau (II.2).

Tableau (II.2) Compléxité de calcul des moments 2D de Meixner MM_n de l'ordre $(n+m)$

Méthodes	**Multiplications**	**Additions**
Méthode directe	4NM-4	2NM-M-3
Formule de Clenshaw	2NM+3M+1	2NM+2M

3.2. Reconstruction d'image 2D par les moments de Meixner via la formule de Clenshaw

La récursive à trois termes des polynômes pondéré de Meixner est donné par:

$$\tilde{M}_{n+1}^{(\gamma,\mu)}(x)=\frac{x(\mu-1)+\mu(n+\gamma)+n}{\mu}\sqrt{\frac{\mu}{(n+1)(n+\gamma)}}\tilde{M}_n^{(\gamma,\mu)}(x)$$
$$-\frac{n(n-1+\gamma)}{\mu}\sqrt{\frac{\mu^2}{n(n+1)(n+\gamma)(n+\gamma-1)}}\tilde{M}_{n-1}^{(\gamma,\mu)}(x) \qquad \textbf{(II.38)}$$

Dont les initiales sont :

$$\tilde{M}_0^{(\gamma,\mu)}(x)=\sqrt{\frac{\omega(x)}{\rho(0)}}$$
$$\tilde{M}_1^{(\gamma,\mu)}(x)=\left(\gamma+x-\frac{x}{\mu}\right)\sqrt{\frac{\mu}{\gamma}}\tilde{M}_0^{(\gamma,\mu)}(x) \qquad \textbf{(II.39)}$$

Pour évaluer l'équation (II.18) en utilisant la formule de récurrence de Clenshaw, nous considérons :

$$F_n(x) = \tilde{M}_n^{(\gamma,\mu)}\left(x\right) \tag{II.40}$$

Comparant (II.22) avec (II.38), nous avons :

$$\begin{aligned}
\alpha_b\left(n,x\right) &= \frac{x\left(\mu-1\right)+\mu\left(n+\gamma\right)+n}{\mu}\sqrt{\frac{\mu}{\left(n+1\right)\left(n+\gamma\right)}} \\
\beta_b\left(n,x\right) = \beta_b\left(n\right) &= -\frac{n\left(n-1+\gamma\right)}{\mu}\sqrt{\frac{\mu^2}{n\left(n+1\right)\left(n+\gamma\right)\left(n+\gamma-1\right)}}
\end{aligned} \tag{II.41}$$

En appliquant la formule de récurrence de Clenshaw sur l'équation (II.18), nous aurons:

$$\begin{aligned}
f\left(x\right) &= MM_0F_0(x) + F_1(x)\psi_1 + \beta_b(1)F_0(x)\psi_2 \\
&= MM_0\tilde{M}_0^{(\gamma,\mu)}\left(x\right) + \tilde{M}_1^{(\gamma,\mu)}\left(x\right)\psi_1 + \beta_b(1)\tilde{M}_0^{(\gamma,\mu)}\left(x\right)\psi_2
\end{aligned} \tag{II.42}$$

Notez que

$$\tilde{M}_1^{(\gamma,\mu)}\left(x\right) = \alpha_b\left(0,x\right)\tilde{M}_0^{(\gamma,\mu)}\left(x\right)$$

où

$$\alpha_b\left(0,x\right) = \left(\gamma + x - \frac{x}{\mu}\right)\sqrt{\frac{\mu}{\gamma}} \qquad \text{et} \qquad \beta_b(1) = -\frac{\gamma}{\mu}\sqrt{\frac{\mu^2}{2\gamma\left(1+\gamma\right)}}$$

Selon l'équation (II.41), donc :

$$f\left(x\right) = \tilde{M}_0^{(\gamma,\mu)}\left(x\right)\left[MM_0 + \left(\gamma + x - \frac{x}{\mu}\right)\sqrt{\frac{\mu}{\gamma}}\psi_1 - \frac{\gamma}{\mu}\sqrt{\frac{\mu^2}{2\gamma\left(1+\gamma\right)}}\psi_2\right] \tag{II.43}$$

où ψ_1 et ψ_2 sont calculé récurssivement à partir de :

$$\begin{aligned}
&\psi_{N+1} = \psi_N = 0 \\
&\psi_n = \alpha_b\left(n,x\right)\psi_{n+1} + \beta_b\left(n+1\right)\psi_{n+2} + MM_n
\end{aligned} \tag{II.44}$$

pour $n = N-1,....,1$, selon (II.44) avec *n=0*, nous obtenons :

$$\begin{aligned}\psi_0 &= \alpha_b(0,x)\psi_1 + \beta_b(1)\psi_2 + MM_0 \\ &= MM_0 + \left(\gamma + x - \frac{x}{\mu}\right)\sqrt{\frac{\mu}{\gamma}}\psi_1 - \frac{\gamma}{\mu}\sqrt{\frac{\mu^2}{2\gamma(1+\gamma)}}\psi_2\end{aligned} \quad \textbf{(II.45)}$$

La formule (II.43) peut être écrite :

$$f(x) = \psi_0 \tilde{M}_0^{(\gamma,\mu)}(x) \quad \textbf{(II.46)}$$

Dans la méthode récursive proposée pour calculer la transformée des moments de Meixner inverses, nous avons besoin de 2N + 1 multiplications et de 2N additions pour chaque valeur de x. Pour calculer la transformée inverse des moments de Meixner donnée dans l'équation (II.18) En utilisant la méthode directe, nous devons calculer les polynômes de Meixner pondérés jusqu'à l'ordre N-1 en utilisant la formule polynomiale récursive donnée dans (II.38), qui nécessite 3N-4 multiplications, y compris la multiplication avec la racine carrée de la fonction de poids et N-2 additions pour chaque valeur de x. l'Evaluation de la somme donnée en (II.18) nécessite N multiplications et N-1 additions. Par conséquent, pour le calcul de la transformée inverse des moments de Meixner pour une valeur unique de x, il faut un total de (3N-4)+N=4N-4 multiplications et de (N-2)+(N-1)=2N-3 additions. La même chose est donnée dans le tableau (II.3).

Tableau (II.3) : Compléxité de calcul des moments inverse $1D$ Meixner

Methods	**Multiplications**	**Additions**
Méthode directe	4N-4	2N-3
Formule de Clenshaw	2N+1	2N

Semblable au principe de dérivation de l'algorithme de reconstruction de signal 1D, nous pouvons obtenir l'algorithme de reconstruction d'image 2D via la transformée inverse des moments de Meixner. En raison de la séparabilité des polynômes de Meixner 2D, la formule de reconstruction (II.19) peut être réécrite comme suit :

$$f(x,y) = \sum_{n=0}^{M-1} \omega_n(y)\tilde{M}_n^{(\gamma,\mu)}(x) \quad \textbf{(II.47)}$$

avec

$$\omega_n(y) = \sum_{m=0}^{N-1} MM_{nm} \tilde{M}_m^{(\gamma,\mu)}(y) \tag{II.48}$$

Ainsi, les coefficients $\omega_n(y)$ définis dans l'équation (II.48) sont initialement évalués sur l'axe des 'y' pour $(n = M-1,....,1,0)$:

$$\begin{aligned}
&\psi_{N+1} = \psi_N = 0 \\
&\psi_m = \alpha_b(m,y)\psi_{m+1} + \beta_b(m+1)\psi_{m+2} + MM_{nm} \\
&\qquad (m = N-1,....,1,0), \\
&\omega_n(y) = \psi_0 \tilde{M}_0^{(\gamma,\mu)}(y)
\end{aligned} \tag{II.49}$$

Puis, les coefficients $\omega_n(y)$ sont utilisés pour évaluer la fonction $f(x,y)$ définie dans (II.47) sur l'axe des 'y':

$$\begin{aligned}
&\varphi_{M+1} = \varphi_M = 0 \\
&\varphi_n = \alpha_b(n,x)\varphi_{n+1} + \beta_b(n+1)\varphi_{n+2} + \omega_n(y) \\
&\qquad (n = M-1,....,1,0), \\
&f(x,y) = \varphi_0 \tilde{M}_0^{(\gamma,\mu)}(x)
\end{aligned} \tag{II.50}$$

Dans l'équation (II.49), l'évaluation de $\omega_n(y)$ nécessite 2N+1 multiplications et 2N additions pour chaque valeur de n. Comme il existe M valeurs de n, un total de 2NM+M multiplications et 2NM additions sont nécessaires. L'équation (II.50) nécessite 2M+1 multiplications et 2M additions. Par conséquent, le calcul de $f(x,y)$ en se basant sur l'implémentation récursive proposée dans (II.49) et (II.50) nécessite un total de 2MM+3M+1 multiplications et de 2NM + 2M additions pour chaque valeur de (x,y). La méthode directe pour calculer la transformée inverse des moments de Meixner en 2D se fait en deux étapes, comme indiqué dans (II.47) et (II.48), qui nécessite (4N-4)M+4M-4=4NM-4 multiplications et (2N-3)M+2M-3=2NM-M-3 additions. La même chose est donnée dans le tableau (II.4)

Tableau (II.4): Complexité du calcul des moments inverse $2D$ de Meixner

Méthodes	Multiplications	Additions
Méthode directe	4NM-4	2NM-M-3
Formule de Clenshaw	2NM+3M+1	2NM+2M

Dans la suite, nous donnons des résultats expérimentaux pour valider les résultats théoriques développés précédemment. Cette partie est divisée en trois sous-sections. Dans la première sous-section, nous allons comparer le calcul temporel des moments de Meixner selon trois méthodes: la première est la relation de récurrence par rapport à n, la deuxième une relation de récurrence par rapport à x et la troisième est la méthode proposée pour les images 2D de niveaux binaire et de niveaux de gris. Dans la seconde sous-section, nous avons également comparé le temps de calcul de la reconstruction. Dans la dernière sous-section, nous testerons la capacité des moments de Meixner à reconstituer des images binaires et en niveaux de gris avec et sans bruit.

4. Temps de calcul des moments de Meixner

Des algorithmes rapides sont souhaitables dans une large gamme d'applications de traitement d'images et de vision par ordinateur, en particulier pour les applications en temps réel et les images de grande taille. Ces algorithmes rapides sont basés sur la réduction des temps de calcul. Afin de garantir l'efficacité de la méthode proposée, plusieurs expériences numériques sont menées. Notez que toutes nos expériences numériques ont été effectuées à l'aide d'un PC doté d'un processeur Intel® Core i5-4200M à 1,60 GHz de fréquence et 4,00 Go de RAM et exploités par Windows 10 Professionnel 64 bits. Le code exécuté est conçu avec Matlab8.

Dans cette sous-section, nous allons comparer le temps de calcul des moments de Meixner selon trois méthodes: la relation de récurrence en trois termes par rapport à la variable n et x basées sur les équations (II.10), (II.14) et l'application de l'équation (II.17) ainsi que la méthode basée sur la formule de récurrence de Clenshaw définie précédemment par (II.37).

Le taux d'amélioration du temps d'exécution (ETIR) est utilisé comme critère [Hosny, 2012]. Ce rapport est défini comme ETIR% = (1 − Time1 / Time2) × 100, où Time1 et Time2 correspondent au temps d'exécution des méthodes proposées et existantes. Il est clair que ETIR = 0 si les deux temps d'exécution sont identiques. Nous avons seulement montré le critère ETIR de la méthode proposée par rapport à la méthode récursive dans [Koekoe, 2010] et la méthode récursive dans [Zhu, 2010] en utilisant les symboles ETIRPR1 et ETIRPR2, respectivement.

Dans le premier exemple, un ensemble de trois images binaires de dimensions 128×128 (Figure II.1) sélectionnées dans la base de données bien connue MPEG-7 CE-shape-1 Partie B [MPEG-7, 2014], ont été utilisés comme image de test. Les processus de calcul sont exécutés 10 fois pour les moments de Meixner pour chacune des trois images, où les temps moyens des moments de Meixner et le taux d'amélioration du temps d'exécution (ETIR) sont inclus dans le tableau (II.5). Les résultats obtenus sont représentés et affichés dans le tableau (II.5). Le résultat obtenu indique que la méthode proposée présente de meilleures performances que la relation de récurrence en respect à n et x en termes de temps de calcul des moments de Meixner. Les temps de calcul moyens sont tracés en fonction de l'ordre des moments sur la figure (II.2).

Dans le deuxième exemple, un ensemble de trois images en niveaux de gris de dimensions 150×150, illustrées sur la figure (II.3), ont été utilisé. Les processus de calcul sont effectués 10 fois sur les moments de Mexiner pour chacune des trois images, où les temps moyens des moments de Meixner et le taux d'amélioration du temps d'exécution (ETIR) sont inclus dans le tableau (II.6). Le résultat indique également que la méthode rapide proposée présente de meilleures performances que la relation de récurrence en respect à n et x, en termes de temps de calcul des moments de Mexiner. Le temps de calcul moyen est tracé par rapport à l'ordre des moments sur la figure (II.4).



Figure (II.1): Ensemble d'images binaires de dimension 128×128 pixels: Tree, Children et Deer

Tableau (II.5): Temps moyens et rapport ETIR pour le calcul des moments d'ordres différents pour des images binaires

Order	Recursive method [9]	Recursive method [30]	Proposed Method	$ETIR_{PR1}$	$ETIR_{PR2}$
5	0.0066	0.0043	0.0042	38.5863%	1.0364%
10	0.0138	0.0091	0.0056	57.4645%	36.7295%
20	0.0434	0.0292	0.0121	77.5523%	66.2406%
30	0.0910	0.0608	0.0140	84.4192%	76.3478%
40	0.1608	0.1053	0.0189	87.9777%	81.9912%
50	0.2517	0.1619	0.0239	90.2394%	85.3874%
60	0.3849	0.2315	0.0284	90.9025%	87.7259%
70	0.4684	0.3107	0.0365	92.7252%	88.9588%
80	0.6020	0.3985	0.0409	93.4447%	89.8868%
90	0.7649	0.5075	0.0469	94.1488%	91.0084%
100	0.9754	0.6387	0.0539	94.4288%	91.6778%

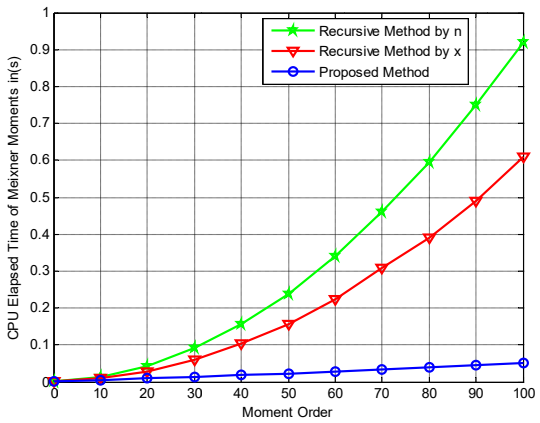


Figure (II.2): Temps moyens (CPU) nécessaire pour le calcul des moments de Meixner de différents ordres pour des images binaires



Figure (II.3): Ensemble d'images de niveaux de gris de dimensions 150×150 pixels: House, Pirate et Lake

Tableau (II.6) : Temps moyens et rapport ETIR pour le calcul des moments d'ordres différents pour des images de niveaux de gris

Order	Recursive method [9]	Recursive method [30]	Proposed Method	$ETIR_{PR1}$	$ETIR_{PR2}$
5	0.0094	0.0066	0.0057	39.7808%	13.1685%
10	0.0190	0.0123	0.0076	58.0811%	38.9242%
20	0.0607	0.0414	0.0131	78.3653%	67.8924%
30	0.1289	0.0820	0.0192	86.7362%	77.2367%
40	0.2732	0.1520	0.0309	87.6578%	81.3485%
50	0.3626	0.2300	0.0328	91.1087%	85.5717%
60	0.5020	0.3300	0.0404	91.9927%	87.7751%
70	0.7584	0.4573	0.0460	92.2215%	89.5703%
80	0.8496	0.5660	0.0533	93.5174%	90.4986%
90	1.0806	0.6927	0.0622	94.1154%	91.1413%
100	1.3035	0.8474	0.0687	94.6816%	91.9002%

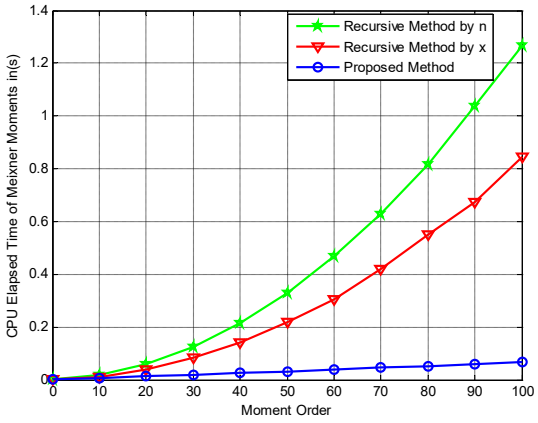


Figure (II.4): Temps moyens (CPU) nécessaire pour le calcul des moments de Meixner de différents ordres pour des images de niveaux de gris

5. Temps de calcul de la transformée inverse des moments de Meixner

Les temps CPU écoulés (Elapsed CPU times) sont utilisés pour montrer l'efficacité de la méthode proposée lorsque ces temps écoulés sont calculés à l'aide de la méthode proposée et de la méthode existante [Koekoek, 2010] and [Zhu, 2010] dans le même environnement informatique. Dans cette sous-section, nous allons comparer le temps de calcul pour la reconstruction des images binaires et en niveaux de gris par trois méthodes: la relation de récurrence par rapport à la variable n et x basées sur les équations (II.10), (II.14), et l'application de l'équation (II.19) ainsi que par la formule de récurrence de Clenshaw définie précédemment par (II.50).

Dans la première expérience, un ensemble d'images binaires Bird, Device et Pocket de dimensions 128×128 pixels (Figure II.5) sélectionnées dans la base de données bien connue MPEG-7 CE-shape-1 Part B [MPEG-7, 2014] ont été utilisés comme images de tests. Les processus de calcul sont effectués 10 fois pour chacune des trois images, où les temps moyens et le rapport ETIR sont inclus dans le tableau (II.7). Les temps moyens écoulés sont représentés par rapport à l'ordre des moments sur la figure (II.6). Il est clair que les méthodes de récurrence de n et x prennent beaucoup de temps et ne conviennent pas aux applications et systèmes en ligne où elles nécessitaient un temps d'exécution plus long. De l'autre côté, la méthode proposée est une méthode très rapide.

Dans la deuxième expérience numérique, un ensemble d'images en niveaux de gris ([0,255]) Highway, Peppers et Insidecity de dimensions 150×150 est utilisé. La figure (II.7) illustre ces images. Les processus de calcul sont exécutés et répétés 10 fois pour chacune des 3 images, où les temps moyens écoulés de la CPU et le rapport d'amélioration du temps d'exécution (ETIR) sont inclus dans le tableau (II.8). Les temps moyens écoulés sont représentés par rapport à l'ordre des moments sur la figure (II.8). Il est clair que la méthode proposée a considérablement réduit le temps d'exécution.



Figure (II.5): Ensemble d'images binaires de dimensions 128×128 pixels: Bird, Device et Pocket

Tableau (II.7) : Temps moyens et rapport ETIR pour le calcul des moments inverses d'ordres différents pour des images binaires

Order	Recursive method [9]	Recursive method [30]	Proposed Method	$ETIR_{PR1}$	$ETIR_{PR2}$
5	0.0298	0.0197	0.0142	51.8208%	27.6877%
10	0.0580	0.0379	0.0177	66.8600%	50.1306%
20	0.1495	0.1036	0.0266	79.9106%	74.9963%
30	0.2891	0.2037	0.0356	87.5789%	82.8537%
40	0.4758	0.3205	0.0444	90.5763%	86.1702%
50	0.7102	0.4728	0.0533	92.6131%	88.7812%

60	0.9639	0.6452	0.0613	93.3264%	90.4849%
70	1.3159	0.8565	0.0740	94.3727%	91.6365%
80	1.6546	1.1115	0.0845	94.7634%	92.4186%
90	2.0926	1.3930	0.0930	95.3361%	93.2545%
100	2.5612	1.7374	0.1105	95.5923%	93.8800%

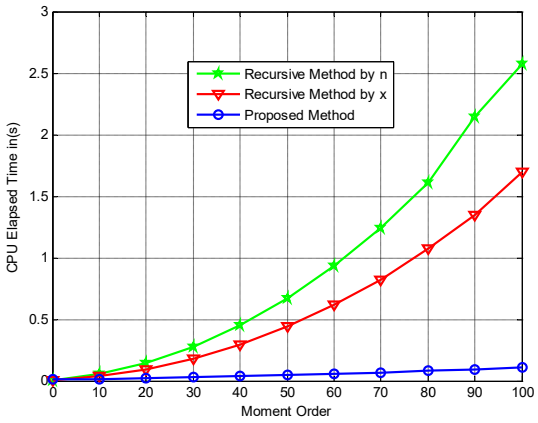


Figure (II.6): Temps moyens (CPU) nécessaire pour le calcul des moments de Meixner inverses de différents ordres pour des images binaires



Figure (II.7): Ensemble d'images de niveaux de gris de dimensions 150×150 pixels: House, Pirate et Lake

Tableau (II.8): Temps moyens et rapport ETIR pour le calcul des moments inverses d'ordres différents pour des images de niveaux de gris

Order	Recursive method [9]	Recursive method [30]	Proposed Method	$ETIR_{PR1}$	$ETIR_{PR2}$
5	0.0416	0.0275	0.0191	53.5434%	24.8550%
10	0.0747	0.0499	0.0237	68.5530%	52.1663%
20	0.1964	0.1311	0.0337	83.6488%	74.1622%
30	0.3910	0.2602	0.0466	88.3564%	81.1462%
40	0.6218	0.4263	0.0562	91.2885%	86.7164%
50	0.9710	0.6461	0.0740	92.6012%	89.2077%

60	1.3140	0.8798	0.0810	94.0354%	90.7814%
70	1.8851	1.1969	0.1075	94.4918%	91.9324%
80	2.3149	1.4719	0.1072	94.9599%	92.5475%
90	2.7850	1.8580	0.1228	95.5433%	93.4147%
100	3.4613	2.3189	0.1427	96.0706%	94.2481%

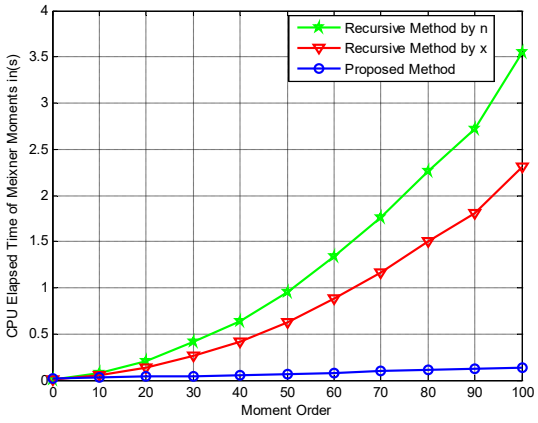


Figure (II.8): Temps moyens (CPU) nécessaire pour le calcul des moments de Meixner inverses de différents ordres pour des images de niveaux de gris

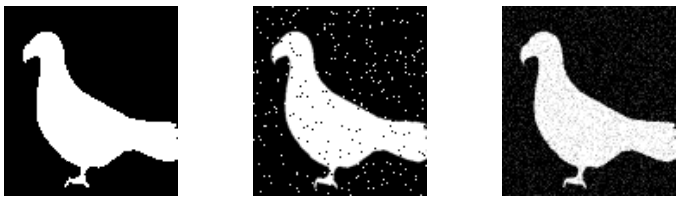
6. Reconstruction d'image 2D par les moments de Meixner

La précision de la méthode proposée est démontrée en utilisant l'aspect de reconstruction d'image pour les images sans bruit et les images contaminées par différents types de bruit, sont discutées dans cette sous-section. Nous discuterons de la capacité des moments de Meixner pour la reconstruction des images binaires et en niveaux de gris en utilisant la méthode proposée. Pour évaluer cette méthode, nous allons calculer l'erreur au carré principale (MSE) définie par (II.20) entre l'image originale et l'image reconstruite. La MSE est largement utilisée dans le domaine de l'analyse d'images en tant que mesure quantitative de la précision.

Deux expériences numériques sont effectuées pour vérifier la capacité de reconstruction d'image du procédé proposé, où l'image binaire 'Bird' de dimensions 128×128, illustrée dans la figure (II.5), est utilisée avec un ordre de moments maximal compris entre 0 et 100 et les valeurs de gris de 0 à 255. L'image à l'échelle *"Poivrons"* de taille 150×150, est

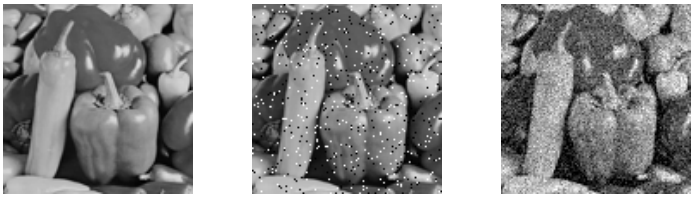
utilisé (Figure II.7) avec un ordre de moment maximal compris entre 0 et 100. Les Figures (II.11) et (II.12) montrent la MSE du procédé proposé. Il est clair que, la MSE décroît à mesure que l'ordre du moment augmente, la MSE approche de zéro de plus en plus. L'image reconstruite sera très proche de celle d'origine lorsque l'ordre du moment maximum atteint une certaine valeur. La résistance au bruit par les moments de Meixner est une propriété très souhaitable. Pour garantir la robustesse de ces moments aux effets négatifs de différents types de bruit, deux expériences numériques sont conduites à l'aide d'images contaminées par le bruit. Deux types de bruit sont utilisés dans ces expériences numériques. Le premier type de bruit est le bruit *'sel et poivre'*, tandis que le second est le bruit *'blanc gaussien'*. Les images contaminées avec deux types de bruit sont affichées sur les Figures (II.9.b), (II.9.c) et les Figures (II.10.b), (II.10.c) respectivement. Les quatre images contaminées sont reconstruites en utilisant les moments Meixner.

Les courbes tracées de MSE pour les images contaminées par le bruit sont représentées dans (II.11) et (II.12). Il convient de noter que les trois courbes de MSE sont tracées sur la même figure pour faciliter la comparaison. En règle générale, les valeurs de la MSE des images bruitées sont supérieures à leurs valeurs MSE correspondantes de l'image sans bruit. Comme le montrent les figures (II.11) et (II.12), les courbes MSE des images contaminées par le bruit s'approchent de zéro en augmentant l'ordre des moments. Les résultats de ces expériences garantissent la robustesse des moments de Meixner contre les différents types de bruit.



(a) (b) (c)

Figure (II.9): image binaire « Bird »: (a) noise-free image (sans bruit), (b) noisy image: Salt & Peppers (bruit sel et poivre), (c) noisy image: white Gaussian (bruit Gaussian blanc)



(a) (b) (c)

Figure (II.10): image de niveaux de gris « Pepper »: (a) noise-free image (sans bruit), (b) noisy image: Salt & Peppers (bruit sel et poivre), (c) noisy image: white Gaussian (bruit Gaussian blanc)

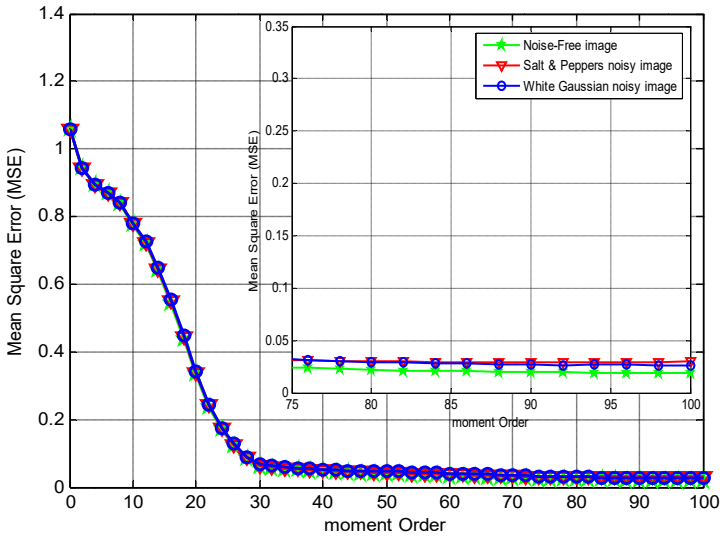


Figure (II.11): MSE de l'image binaire reconstruite « bird ».

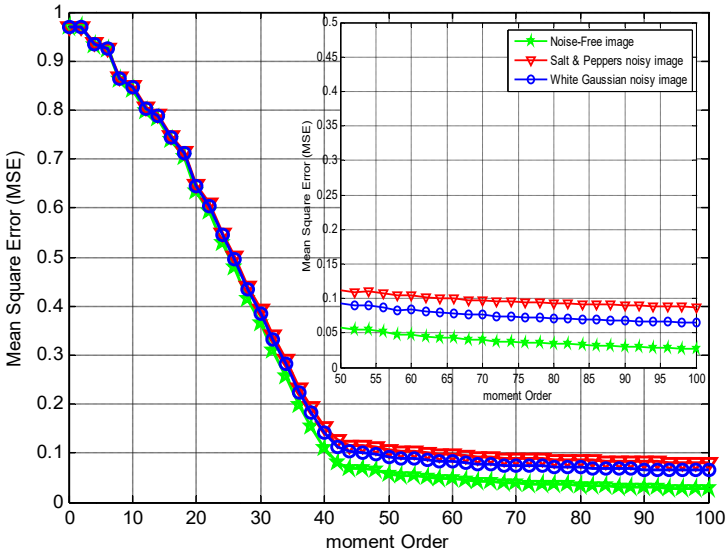


Figure (II.12): MSE de l'image reconstruite en niveaux de gris « Pepper »

Les figures (II.13) et (II.14) montrent les résultats de la reconstruction d'images de niveaux binaires de poivrons et binaires d'un oiseau avec et sans bruit par la méthode proposée. Moments jusqu'à l'ordre maximum 8, 16, 20, 40, 60 et 10, 30, 50, 70, 90, respectivement. Les résultats de ces expériences montrent la faisabilité de la méthode proposée pour le calcul du moment de Meixner et de sa transformée inverse.

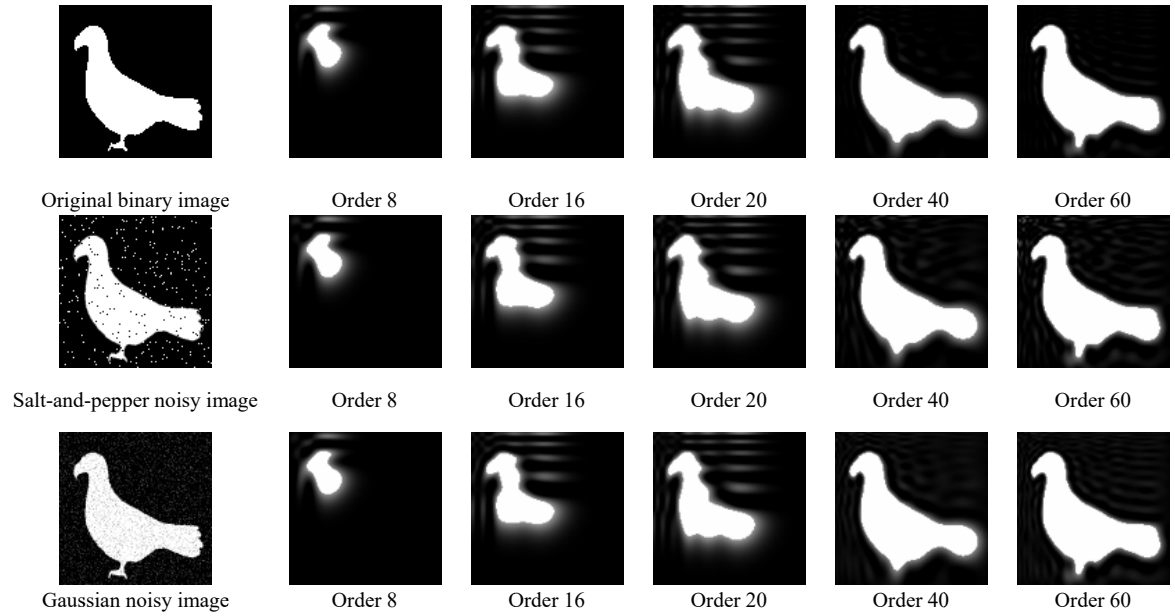


Figure (II.13): Reconstruction d'image binaire sans et avec bruit pour des ordres différents

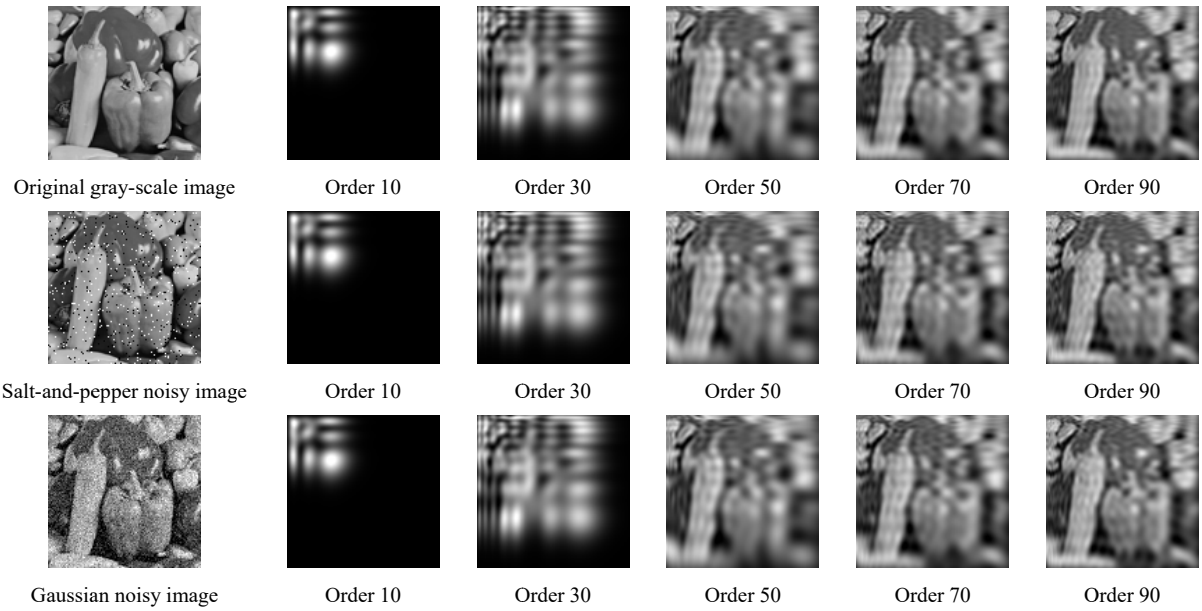


Figure (II.14): Reconstruction d'image de niveaux de gris sans et avec bruit pour des ordres différents

7. Conclusion

Dans ce chapitre, nous avons présenté un algorithme rapide pour calculer les moments de Meixner et sa transformée inverse pour les images binaires et à niveaux de gris à l'aide de la formule de récurrence de Clenshaw.

Cette méthode rapide est réalisée en dérivant certaines propriétés des polynômes orthogonaux discrets de Meixner. La méthode proposée nécessite moins de calcul que la méthode simple; elle peut réduire considérablement le temps de calcul des moments de Meixner et de sa transformée inverse. L'efficacité des moments de Meixner proposés pour la reconstruction d'images binaires et en niveaux de gris avec bruit et sans bruit a été démontrée par des résultats expérimentaux garantissant la robustesse des moments de Meixner contre les différents types de bruit.

CHAPITRE 3 : CALCUL DES MOMENTS DE MEIXNER ET LEURS INVERSE PAR LA METHODE DES BLOCS VIA LES FILTRES NUMERIQUES

Dans ce chapitre, nous proposons une nouvelle méthode pour le calcul rapide des moments orthogonaux discrets de Meixner et de ses inverses. Dans cette méthode, nous avons utilisé la notion de filtres numériques basée sur la transformation en Z, à la fois pour accélérer le temps de calcul de Meixner et pour réduire l'erreur de reconstruction des images. Pour garantir la stabilité numérique et la robustesse vis-à-vis du bruit, nous proposons deux algorithmes qui traitent les images comme un ensemble de blocs dans lesquels chaque bloc sera traité indépendamment. En fait, grâce au premier algorithme, les moments de Meixner sont calculés à partir d'un ensemble de blocs géométriques de taille fixe. Par ailleurs, dans le deuxième algorithme, les images sont représentées par un ensemble de tranches d'image (slice), chaque tranche contenant plusieurs blocs homogènes de tailles différentes. Les moments de Meixner, dans ce cas, sont calculés à partir de chaque bloc de chaque tranche. L'application de ces deux algorithmes nous a permis de déduire une réduction significative de l'information traitée et de l'espace image, ce qui permet d'utiliser des moments de Meixner de faible ordre pour une meilleure description des détails les plus fins des images. Les performances de la méthode proposée sont démontrées à travers plusieurs simulations sur différentes bases d'images.

1. Introduction

L'utilisation de moments orthogonaux discrets basés sur des polynômes orthogonaux discrets élimine le besoin d'approximation numérique et satisfait précisément la propriété d'orthogonalité [yap, 2003].

Ce domaine de recherche a été confronté au coût du calcul et à la qualité de la reconstruction. Ainsi, de nombreux algorithmes ont été implémentés pour le calcul rapide des moments et de leurs inverses. Spiliotis [Spiliotis, 1998] a proposé un algorithme basé sur la représentation en blocs d'une image binaire. Cet algorithme réduit considérablement les fluctuations numériques causées par le calcul des dimensions les plus élevées. Papakostas

[Papakostas, 2010b] a introduit une nouvelle méthode utilisée sur les images en niveaux de gris. Il a proposé une représentation partielle du bloc d'image (PIBR) pour calculer les moments rapidement et efficacement, ce qui a été prouvé par Hmimid pour les moments orthogonaux [Hmimid, 2014].

Wang [Wang, 2006] a utilisé une formule de récurrence de Clenshaw pour accélérer le calcul des moments de Tchebichef. Alors que Honarvar [Honarvar, 2013 et 2014] a introduit le filtre numérique en cascade 1D et 2D utilisant les propriétés de transformée en Z en termes de polynômes de Tchebichef et de Krawtchouk. Les travaux précédents utilisant le filtre numérique ont été introduits par Hatamian [Hatamian, 1986] puis par Wong [Wong, 1999]. Les deux œuvres se concentrent sur des moments géométriques.

Ces méthodes permettent l'obtention d'une excellente qualité d'images reconstruites en réduisant les erreurs. Mais, dans leur étude, seules les images binaires et de petite taille sont prises en compte, mais pas les images en niveaux de gris avec des tailles plus grandes. En effet, pour les images de grandes dimensions, ces méthodes révèlent une instabilité numérique avec une perte importante d'informations. Par exemple, les valeurs des moments d'ordre 40 x 40 vont jusqu'à 2,2571e + 018 pour une image de taille 100 x 100.

Ces difficultés nous ont motivés pour proposer deux algorithmes plus efficaces. Pour le premier algorithme proposé, nous avons utilisé la méthode d'analyse d'image en bloc par recouvrement [Fadili, 2003]. Cette méthode est purement locale, chaque bloc étant traité indépendamment. Ainsi, la réduction de l'espace d'image permet l'utilisation de moments orthogonaux discrets de faibles ordres pendant le processus de calcul. En effet, le calcul des moments orthogonaux discrets de l'image par les blocs offre une qualité amélioré de représentation de l'image. Cependant, le choix de la taille du bloc ne dépend d'aucun critère. La seconde méthode proposée, basée sur la représentation de blocs d'images et de tranches d'images (IBR-ISR), est adoptée pour le calcul de moments orthogonaux discrets. Cette méthode est axée sur la décomposition de l'image en un ensemble de tranches et de blocs homogènes, où les moments de l'image seront calculés à partir de chaque bloc de chaque tranche, au lieu de chaque pixel de l'image. Par conséquent, la quantité d'informations dans chaque bloc est très petite, ce qui permet l'utilisation de moments orthogonaux discrets d'ordres faibles pendant le processus de calcul.

Ainsi, plusieurs simulations ont été réalisées pour démontrer l'efficacité des algorithmes proposés en termes de stabilité numérique, de temps de calcul, de qualité de la reconstruction et de robustesse vis-à-vis du bruit.

Ce chapitre est organisé en plusieurs sections. La première définit les polynômes de Meixner. La seconde introduit le filtre numérique pour accélérer le calcul des moments de Meixner et de leurs inverses. La troisième section présente nos algorithmes proposés. Dans la dernière section, nous avons effectué des tests sur des images binaires et à niveaux de gris.

2. Les moments de Meixners

Les polynômes de Meixner d'ordre *n* sont représentés en utilisant la fonction hypergéométrique comme suit [Hmimid, 2015]:

$$M_n^{(\gamma,\mu)}(x) = (\gamma)_n \, {}_2F_1(-n,-x;\gamma;1-1/\mu) = \sum_{k=0}^{n} \alpha_{k,n}^{(\gamma,\mu)} x^k \qquad \textbf{(III.1)}$$

où $x, n = 0,1,2,...,N-1, \quad N > 0, \quad and \ \gamma \succ 0, \quad 0 \prec \mu \prec 1$ ou la function ${}_2F_1$ est l'hypergéométrique généralisée définie comme:

$${}_2F_1(a,b;c;z) = \sum_{k=0}^{\infty} \frac{(a)_k (b)_k}{(c)_k} \frac{z^k}{k!} \qquad \textbf{(III.2)}$$

Le symbole $(a)_k$ est le symbole de Pochhammer donné par:

$$(a)_k = a(a+1)(a+2)....(a+k-1) = \frac{\Gamma(a+k)}{\Gamma(a)} \qquad \textbf{(III.3)}$$

Avec l'identité ci-dessus et la définition de la fonction hypergéométrique dans (III.2), le polynôme de Meixner orthogonal dans (III.1) peut également s'écrire comme suit:

$$M_n^{(\gamma,\mu)}(x) = \sum_{k=0}^{n} \frac{(-n)_k (-x)_k}{(\gamma)_k \, k!} \left(1 - \frac{1}{\mu}\right)^k \qquad \textbf{(III.4)}$$

Nous démontrons la relation entre les polynômes de Meixner et les monômes. Tout d'abord, en développant le symbole de Pochhammer en termes de coefficients binomiaux, nous obtenons:

$$(-x)_k = (-1)^k \, k! \binom{x}{k} \tag{III.5}$$

Une forme plus explicite est définie comme suit :

$$M_n^{(\gamma,\mu)}(x) = \sum_{k=0}^{n} \frac{(-n)_k}{(\gamma)_k} \left(\frac{1}{\mu} - 1 \right)^k \binom{x}{k} \tag{III.6}$$

Cette dernière (III.6) peut être exprimé en terme de binomial $\binom{x}{k}$ par:

$$M_n^{(\gamma,\mu)}(x) = \sum_{k=0}^{n} \Gamma_k(n,\gamma,\mu) \binom{x}{k} \tag{III.7}$$

où $\Gamma_k(n,\gamma,\mu)$ est une matrice de triangle inférieure:

$$\Gamma_k(n,\gamma,\mu) = \frac{(-n)_k}{(\gamma)_k} \left(\frac{1}{\mu} - 1 \right)^k \tag{III.8}$$

Les polynômes orthogonaux discrets de Meixner $M_n^{(\gamma,\mu)}(x)$ satisfont une relation orthogonale de la forme:

$$\sum_{x=0}^{N} M_n^{(\gamma,\mu)}(x) M_m^{(\gamma,\mu)}(y) \omega(x) = \rho(n) \delta_{nm} \tag{III.9}$$

avec $\omega(x)$ est la fonction de poids des polynômes orthogonaux discrets de Meixner:

$$\omega(x) = \frac{\mu^x \Gamma(\gamma + x)}{\Gamma(1+x)\Gamma(\gamma)} \tag{III.10}$$

et $\rho(n)$ est la norme au carré des polynômes orthogonaux discrets de Meixner définis par:

$$\rho(n) = \frac{n!(\gamma)_n}{\mu^n (1-\mu)^\gamma} \tag{III.11}$$

Pour éviter les fluctuations dans le calcul numérique, nous utilisons la forme normalisée de polynômes de Meixner définie par:

$$\tilde{M}_n^{(\gamma,\mu)}(x) = M_n^{(\gamma,\mu)}(x)\sqrt{\frac{\omega(x)}{\rho(n)}} \tag{III.12}$$

Pour calculer les moments normalisés de Meixner, nous mettons:

$$MM_{nm} = \sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)f(x,y) \tag{III.13}$$

que nous pouvons exprimer comme suit:

$$MM_{nm} = \frac{1}{\sqrt{\rho(n)\rho(m)}}\sum_{x=0}^{N-1}\sum_{y=0}^{M-1}M_n^{(\gamma,\mu)}(x)M_m^{(\gamma,\mu)}(y)\sqrt{\omega(x)\omega(y)}f(x,y) \tag{III.14}$$

Plus tard pour une certaine raison, nous pouvons la remplacer par la forme suivante:

$$MM_{nm} = A_{nm}\sum_{x=0}^{N-1}\sum_{y=0}^{M-1}M_n^{(\gamma,\mu)}(x)M_m^{(\gamma,\mu)}(y)\tilde{f}(x,y) \tag{III.15}$$

où:

$$A_{nm} = \frac{1}{\sqrt{\rho(n)\rho(m)}} \tag{III.16}$$

$$\tilde{f}(x,y) = \sqrt{\omega(x)\omega(y)}f(x,y) \tag{III.17}$$

3. Calcul rapide des moments de Meixner par le filtre numérique

Dans cette section, nous montrons d'abord qu'une sortie de filtre numérique à rétroaction en cascade peut être échantillonnée aux intervalles de temps précédents. Dans la deuxième sous-section, nous montrons comment calculer correctement les moments de Meixner en termes de sorties de filtres numériques en cascade. Enfin, une représentation matricielle des moments Meixner 2D permettra le calcul direct de ces moments à partir des sorties de filtre.

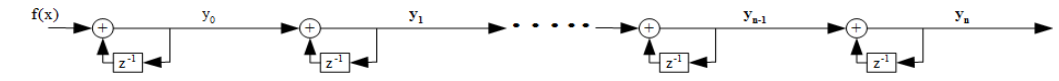
3.1. Les sorties du filtre numérique en cascade

La fonction de transfert d'un filtre en cascade $H_n(z)=\dfrac{1}{\left(1-z^{-1}\right)^{n+1}}$ peut être créé par une série de n accumulateurs en cascade. Comme indiqué sur la figure (III.1.a), un tel filtre, étant donné en entrée le signal f(x), produit une sortie échantillonnée aux points antérieurs de N-n-1 comme suit:

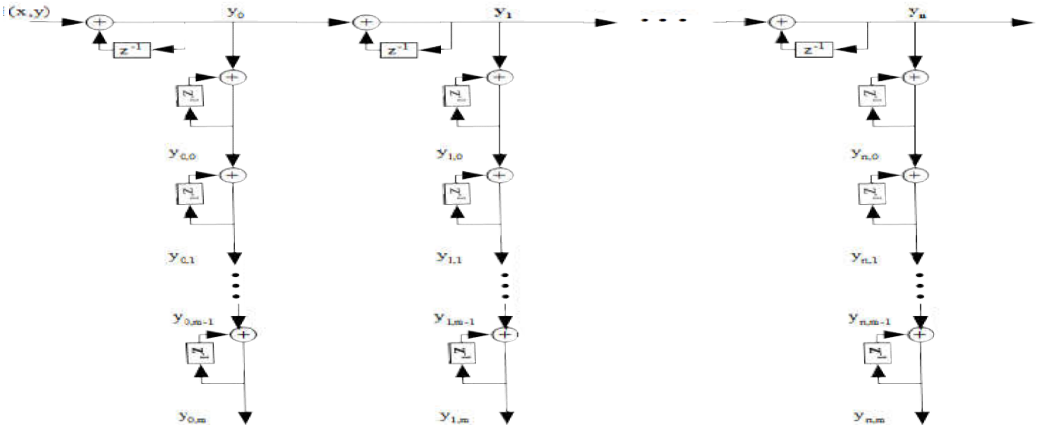
$$y_n(N-n-1)=\sum_{x=0}^{N-1} f(x)\binom{x}{n} \qquad \textbf{(III.18)}$$

Cette formulation permet d'échantillonner les sorties du filtre numérique aux points précédents, N-1, N-2, N-3,…, N-n-1. Pendant ce temps, cet ensemble de valeurs de sortie commence à diminuer après n/2 ordres de moments. Ainsi, une grille d'accumulateur 2D est montrée dans la figure (III.1.b) et consiste en une ligne et n+1 colonnes basées sur des filtres unipolaires à zéro unique. Les sorties de filtre pour une image mise à l'échelle $\tilde{f}(x,y)$ en entrée de taille MxN sont données par l'expression suivante:

$$y_{nm}(N-n-1,M-m-1)=\sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\tilde{f}(x,y)\binom{x}{n}\binom{y}{m} \qquad \textbf{(III.19)}$$



(a)



(b)

Figure (III.1): Structure de filtre numérique pour généré les sorties désirés : (a) y_n de 1D pour (n) ordres. (b) $y_{n,m}$ de 2D pour (n+m) ordre.

3.2. Calcul des moments de Meixner

Dans cette sous-section, une relation directe entre les moments de Meixner et les sorties du filtre numérique est déduite.

En utilisant la définition 2D des moments de Meixner dans (III.15), la forme binomiale des polynômes de Meixner dans (III.7) et les fonctions pondérée et normalisée dans (III.12), nous obtenons l'expression suivante:

$$MM_{nm} = A_{nm} \sum_{r=0}^{n} \sum_{s=0}^{m} \Gamma_r(n,\gamma,\mu)\Gamma_s(m,\gamma,\mu) y_{rs} \tag{III.20}$$

Les moments Meixner générés peuvent être écrits en termes de produit simple:

$$MM_{nm} = A_{nm} Q_{nm} \tag{III.21}$$

où Q est une matrice de NxM, peut être calculé à partir d'un produit matriciel:

$$Q = \Gamma . Y . \Gamma^T \tag{III.22}$$

(T) pour indiqué la transpose d'une matrice, où :

$$MM = \{MM_{nm}\}_{n,m=0}^{N-1,M-1}; Q = \{Q_{nm}\}_{n,m=0}^{N-1,M-1}; Y = \{y_{nm}\}_{n,m=0}^{N-1,M-1} \; and \; \Gamma = \{\Gamma_{nm}\}_{n,m=0}^{N-1,M-1} \tag{III.23}$$

3.3. La transformée inverse des moments de Meixner pour une image 2D

En théorie, une fonction d'image originale *f(x,y)* peut être représentée à partir d'une série infinie de moments de Meixner.

$$f(x,\mathrm{y}) = \sum_{n=0}^{\infty} \sum_{m=0}^{n} MM_{nm} \tilde{M}_n^{(\gamma,\mu)}(x) \tilde{M}_m^{(\gamma,\mu)}(y) \tag{III.24}$$

Mais en pratique, on utilise pour un certain ordre $n \leq T$ que l'on peut calculer, la fonction de l'image d'origine peut être approchée par une série tronquée:

$$\hat{f}(x,\mathrm{y}) = \sum_{n=0}^{T} \sum_{m=0}^{n} MM_{nm} \tilde{M}_n^{(\gamma,\mu)}(x) \tilde{M}_m^{(\gamma,\mu)}(y) \tag{III.25}$$

Plus T approche de l'infini, plus la différence entre l'image originale et l'image reconstruite, tend vers zéro.

La différence entre l'image d'origine $f(x,y)$ et les images reconstruites $\hat{f}(x,y)$ est mesurée en utilisant l'erreur quadratique moyenne (MSE) définie comme suit:

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(f\left(x_i, \mathrm{y}_j\right) - \hat{f}\left(x_i, y_j\right)\right)^2 \quad \textbf{(III.26)}$$

L'utilisation de filtres numériques basés sur la transformée en Z permet de réduire considérablement le temps de calcul des moments orthogonaux discrets de Meixner et de ses inverses. De plus, nous devons également réduire l'erreur de reconstruction. Ainsi, dans la section suivante, nous décrivons deux algorithmes basés sur le partitionnement d'images.

3.4. Méthode de reconstruction d'image 2D par blocs fixes

Dans cette section, nous présentons la méthode d'analyse d'image par blocs fixes [Wang, 2006]. Les moments de Meixner de chaque bloc sont calculés séparément et l'image est reconstruite bloc par bloc.

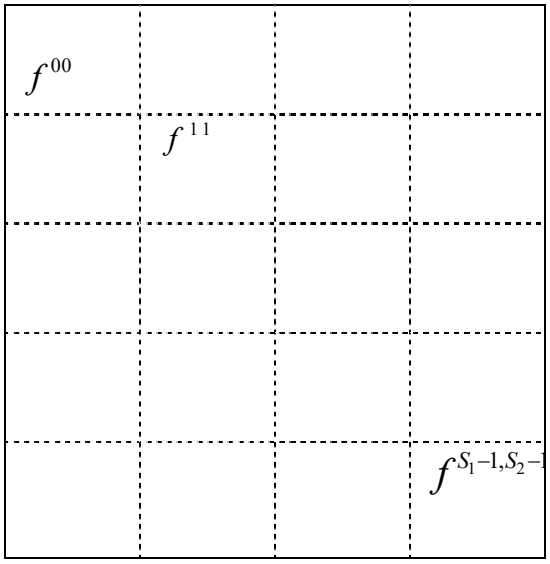
En effet, soit (M,N) la taille de l'image f(x,y), et (k,l) représente la taille du bloc comme indiqué sur la figure (III.2.a), en introduisant les variables:

$$s_1 = \frac{M}{k}, s_2 = \frac{N}{l} \quad \textbf{(III.27)}$$

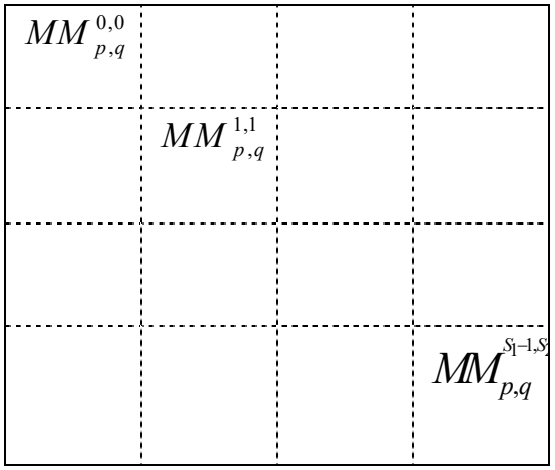
On peut en déduire le nombre total de blocs d'image pouvant être définis comme :

$N_b = s_1 \times s_2$.

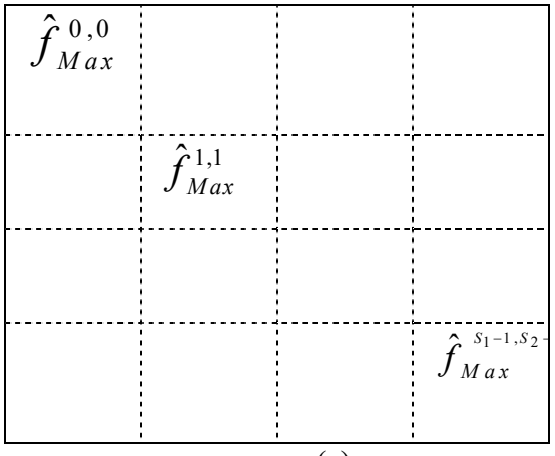
Figure (III.2): Illustration des blocs fixes : (a) Division de l'image originale en Nb sous-images, (b) extractions des moments de chaque bloc, et (c) reconstruction de l'image en fusionnant les Nb blocks à partir de (b).


f^{00}
f^{11}
f^{S_1-1,S_2-1}

(a)


$MM_{p,q}^{0,0}$
$MM_{p,q}^{1,1}$
$MM_{p,q}^{S_1-1,S_2}$

(b)


$\hat{f}_{Max}^{0,0}$
$\hat{f}_{Max}^{1,1}$
$\hat{f}_{Max}^{S_1-1,S_2}$

(c)

La fonction d'image *f(x,y)* est représenté par une ensemble de sous-fonction f^{p_1,p_2} comme sur la figure (III.2.a), alors, on écritv :

$$f^{p_1,p_2}(x,y)=\left\{f(i,j)/\,i,j\in f^{p_1,p_2}\right\} \tag{III.28}$$

Donc, nous pouvons obtenir une image reconstruite comme indiqué dans (Fig.2.c) comme suit :

$$\hat{f}(x,y)=\bigcup_{p_1=0}^{(s_1-1)}\bigcup_{p_2=0}^{(s_2-1)}\hat{f}^{p_1,p_2}(x,y) \tag{III.29}$$

Cela dit, nous introduisons le calcul des moments de Meixner à l'aide d'un filtre numérique sur chaque bloc d'image, comme indiqué dans la figure (III.2.b), nous obtenons alors :

$$MM_{nm}^{p_1,p_2}=A_{nm}\sum_{r=0}^{n}\sum_{s=0}^{m}\Gamma_r(n,\gamma,\mu)\Gamma_s(m,\gamma,\mu)y_{rs}^{p_1,p_2} \tag{III.30}$$

Où :

$$y_{rs}^{p_1,p_2}=\sum_{x=p_2k}^{(p_2+1)k}\sum_{y=p_1l}^{(p_1+1)l}\tilde{f}^{p_1,p_2}(x,y)\begin{pmatrix}x\\r\end{pmatrix}\begin{pmatrix}y\\s\end{pmatrix} \tag{III.31}$$

et $x,y\in f^{p_1,p_2}$.

Les moments de Meixner générés à partir de chaque bloc $MM_{nm}^{p_1,p_2}$, peut être écrits en deux termes d'un produit simple:

$$MM_{nm}^{p_1,p_2}=A_{nm}Q_{nm}^{p_1,p_2} \tag{III.32}$$

Peut être écrit en deux termes d'un produit simple :

$$Q^{p_1,p_2}=\Gamma.Y^{p_1,p_2}.\Gamma^{T} \tag{III.33}$$

(T) dénote la transposition de la matrice.

Et :

$$MM^{p_1,p_2} = \left\{ MM_{nm}^{p_1,p_2} \right\}_{n,m=0}^{N-1,M-1};$$
$$Q^{p_1,p_2} = \left\{ Q_{nm}^{p_1,p_2} \right\}_{n,m=0}^{N-1,M-1};$$
$$Y^{p_1,p_2} = \left\{ y_{nm}^{p_1,p_2} \right\}_{n,m=0}^{N-1,M-1};$$
$$\Gamma = \left\{ \Gamma_{nm} \right\}_{n,m=0}^{N-1,M-1}. \quad \textbf{(III.34)}$$

La figure (III.3), décrit en détail l'algorithme de calcul des moments de Meixner via la répartition d'image 2D en sous blocs de tailles fixes en y impliquant un filtre numéique, et la reconstruction de cette dernière bloc par bloc, à partir des moments calculés sur chaque bloc.

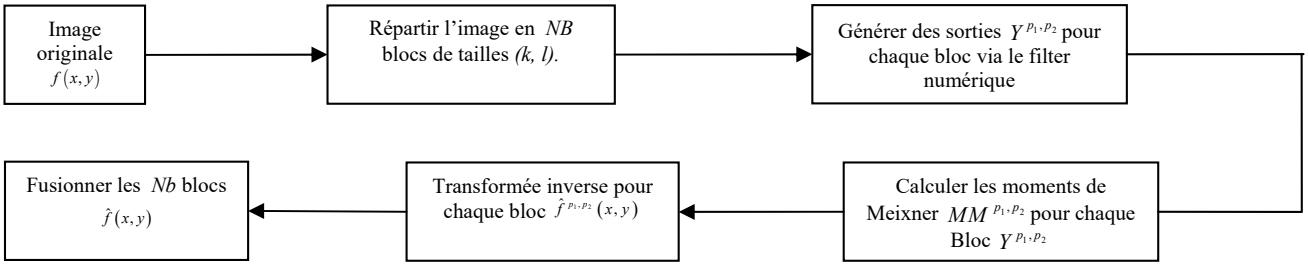


Figure (III.3): Reconstruction d'image par la méthode des blocs fixes

4. Calcul des moments de Meixners et leurs inverse par la méthode IBR-ISR

Dans cette sous-section, nous allons d'abord présenter une brève introduction à la représentation de bloc d'image (IBR) pour les images binaires. Dans la deuxième sous-section, nous donnons une description détaillée de la représentation des tranches d'image (ISR) pour les images en niveaux de gris. Dans la dernière sous-section, nous utiliserons l'algorithme IBR et l'algorithme ISR pour le calcul des moments de Meixner et de leurs inverses pour une image 2D.

4.1. Image binaire

Dans ce cas, l'image binaire est représentée comme un ensemble de blocs. Chaque bloc correspond à un objet ou à une parcelle d'un objet. Ce bloc est défini comme une zone rectangulaire comprenant un groupe de pixels connectés [Preotiuc-Pietro, 2017]:

En appliquant l'algorithme IBR, l'image binaire est décrite comme la relation suivante:

$$f(x,y) = \{b_i, i = 1, 2, ..., K\} \quad \textbf{(III.35)}$$

où bi est le ième bloc et K est le nombre total de blocs, comme montré sur figure (III.4). Chaque bloc est décrit par (x_{1,b_i}, x_{2,b_i}) et (y_{1,b_i}, y_{2,b_i}), ces derniers sont les coordonnées du coin supérieur gauche et du coin inférieur droit dans les axes vertical et horizontal respectivement.

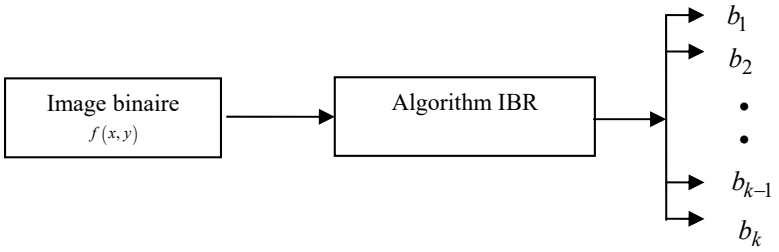


Figure (III.4): Extraction des blocs par l'algorithme IBR

4.2. Image à niveaux de gris

Dans cette section, l'image est décomposée en un certain nombre de tranches binaires et chaque tranche est représentée par un ensemble de blocs [Hmimid, 2014]. Ces blocs sont définis comme une région rectangulaire homogène. En appliquant l'approche ISR, chaque image en niveaux de gris est représentée sous la forme d'un ensemble de tranches binaires $f_i(x,y)$, l'image étant décrite comme la relation suivante:

$$f(x,y) = \sum_{i=1}^{L} f_i(x,y) \quad \textbf{(III.36)}$$

où L est le nombre de tranches et fi la function d'intensité du ième tranche. Dans le cas d'une image binaire L = 1 ainsi $f(x,y) = f_1(x,y)$.

Après avoir décomposé l'image en niveaux de gris en plusieurs tranches, nous pouvons appliquer l'algorithme IBR. L'image en niveaux de gris peut être redéfinie en blocs de différentes intensités comme suit:

$$f(x,y)=\{f_i(x,y), i=1,2,.....,L\}=\{b_{ij}, j=1,2,.....,K_i\} \quad \textbf{(III.37)}$$

$$f_i(x,y)=\{b_{ij}, j=1,2,.....,K_i\} \quad \textbf{(III.38)}$$

où b_{ij} est le bloc du bord i et K_i est le nombre de blocs d'image d'intensité. Chaque bloc est décrit par $(x_{1,b_{ij}}, x_{2,b_{ij}})$ et $(y_{1,b_{ij}}, y_{2,b_{ij}})$, ces derniers sont les coordonnées du coin supérieur gauche et du coin inférieur droit dans les axes vertical et horizontal respectivement.

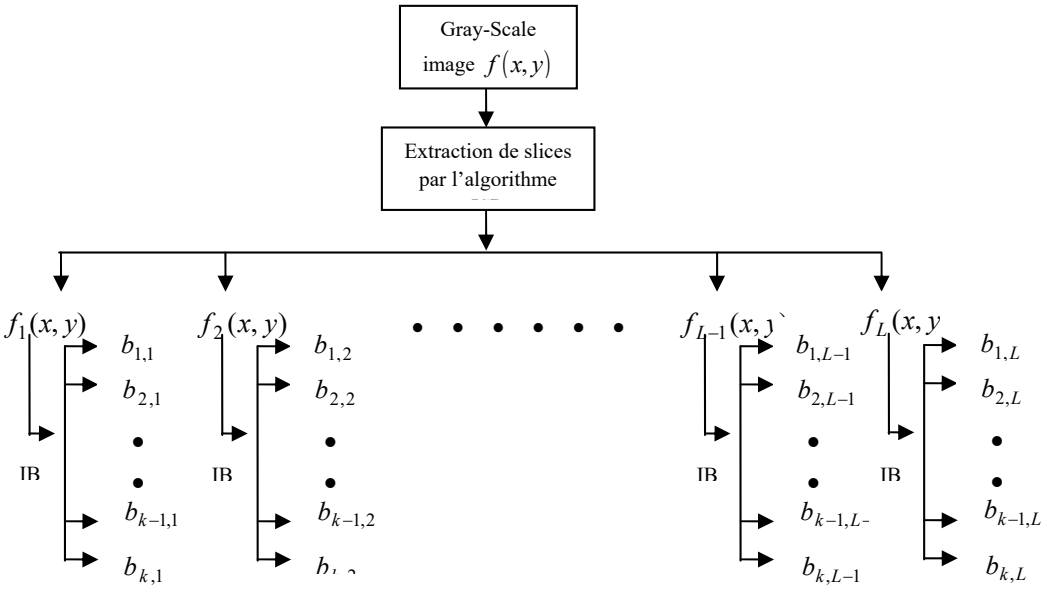


Figure (III.5): Extraction des blocs en combinant les algorithmes ISR-IBR.

Afin d'améliorer la qualité des processus de reconstruction d'image, nous pouvons reconstruire toute l'image avec un nombre fini de moments de Meixner à l'aide de filtres numériques en cascade. Ce fait peut être accompli en reconstruisant chaque bloc séparé en tant que sous-image de taille NixMi, en utilisant la représentation en bloc d'image pour les images binaires (IBR) et la représentation en tranches d'image pour les images en niveaux de gris (IBR-ISR.). Enfin, nous collectons tous la sous-image pour obtenir l'image reconstruite, comme indiqué sur la figure (III.6).

La figure (III.6) décrit le diagramme de calcul des blocs des moments de Meixner pour une image 2D et leurs inverses par IBR-ISR via des filtres numériques.

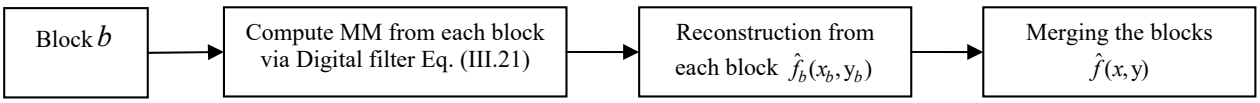


Figure (III.6): Extraction of blocks using ISR algorithm

Pour illustrer notre méthode proposée, prenons comme exemple une fonction $f(x, y)$ de taille 4x4 et de 4 niveaux de gris, comme décrit ci-dessous:

Notre Image discrète est décomposée en 4 blocs de valeurs différentes, de même taille 2x2. Ainsi, nous associons chaque bloc à un filtre numérique et, pour chaque bloc, nous obtenons en sortie un bloc filtré de même taille.

1	1	2	2
1	1	2	2
3	3	4	4
3	3	4	4

Figure (III.7): une fonction discrète f(x,y) de dimensions 4x4

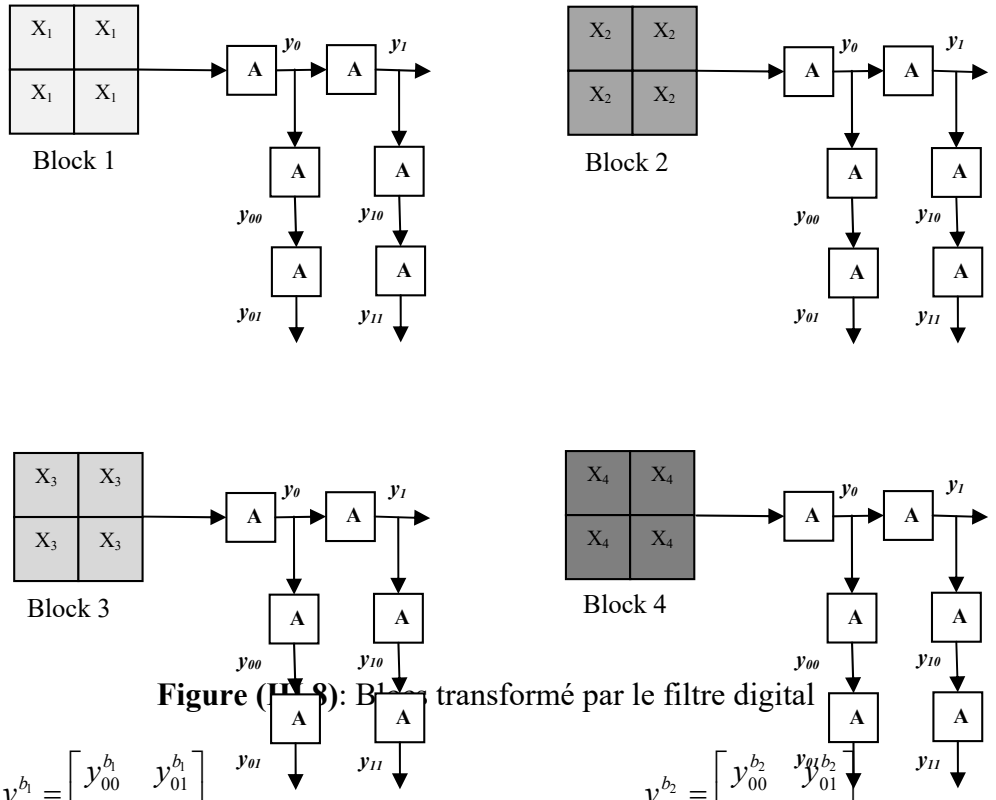


Figure (III.8): Bloc transformé par le filtre digital

$$y_{rs}^{b_1}=\begin{bmatrix} y_{00}^{b_1} & y_{01}^{b_1} \\ y_{10}^{b_1} & y_{11}^{b_1} \end{bmatrix} \qquad y_{rs}^{b_2}=\begin{bmatrix} y_{00}^{b_2} & y_{01}^{b_2} \\ y_{10}^{b_2} & y_{11}^{b_2} \end{bmatrix}$$

$$y_{rs}^{b_3}=\begin{bmatrix} y_{00}^{b_3} & y_{01}^{b_3} \\ y_{10}^{b_3} & y_{11}^{b_3} \end{bmatrix} \qquad y_{rs}^{b_4}=\begin{bmatrix} y_{00}^{b_4} & y_{01}^{b_4} \\ y_{10}^{b_4} & y_{11}^{b_4} \end{bmatrix}$$

Figure (III.9): Outputs of filtred blocks

Ensuite, pour chaque bloc, nous calculons ses moments à l'aide de l'équation (III.21). Ainsi, dans les sorties, nous obtenons pour chaque bloc une matrice de moments:

$$MM_{nm}^{b_1}=\begin{bmatrix} MM_{00}^{b_1} & MM_{01}^{b_1} \\ MM_{10}^{b_1} & MM_{11}^{b_1} \end{bmatrix} \qquad MM_{nm}^{b_2}=\begin{bmatrix} MM_{00}^{b_2} & MM_{01}^{b_2} \\ MM_{10}^{b_2} & MM_{11}^{b_2} \end{bmatrix}$$

$$MM_{nm}^{b_3}=\begin{bmatrix} MM_{00}^{b_3} & MM_{01}^{b_3} \\ MM_{10}^{b_3} & MM_{11}^{b_3} \end{bmatrix} \qquad MM_{nm}^{b_4}=\begin{bmatrix} MM_{00}^{b_4} & MM_{01}^{b_4} \\ MM_{10}^{b_4} & MM_{11}^{b_4} \end{bmatrix}$$

Ensuite, pour reconstruire notre fonction initiale, nous devons reconstruire chaque bloc séparément, en utilisant l'équation (III.24):

$$f^{b_1}(x,y)=\begin{bmatrix} f^{b_1}(0,0) & f^{b_1}(0,1) \\ f^{b_1}(1,0) & f^{b_1}(1,1) \end{bmatrix} \qquad f^{b_2}(x,y)=\begin{bmatrix} f^{b_2}(0,0) & f^{b_2}(0,1) \\ f^{b_2}(1,0) & f^{b_2}(1,1) \end{bmatrix}$$

$$f^{b_3}(x,y)=\begin{bmatrix} f^{b_3}(0,0) & f^{b_3}(0,1) \\ f^{b_3}(1,0) & f^{b_3}(1,1) \end{bmatrix} \qquad f^{b_4}(x,y)=\begin{bmatrix} f^{b_4}(0,0) & f^{b_4}(0,1) \\ f^{b_4}(1,0) & f^{b_4}(1,1) \end{bmatrix}$$

Notre fonction discrète 2D peut être représentée par:

$$f(x,y)=\begin{bmatrix} f^{b1} & f^{b2} \\ f^{b3} & f^{b4} \end{bmatrix}$$

Parmi les trois méthodes ci-dessus, la méthode IBR-ISR fournit un système de calcul de moments orthogonaux discrets plus stable.

Les deux algorithmes proposés dans les deux sections précédentes passent par le filtre numérique pour calculer les moments de Meixner pour une image donnée. Ils sont différents dans la façon dont les blocs sont extraits de l'image: le premier est basé sur la méthode de blocs de tailles fixes (III.3) et le second est basé sur la représentation en tranches et les blocs IBR-ISR (III.6).

5. Simulations et résultats

Dans cette section, nous donnons des résultats expérimentaux pour valider les résultats théoriques développés dans les sections précédentes. Dans la première sous-section, nous comparerons le calcul temporel des moments de Meixner à l'aide des méthodes directe et proposée pour les images binaires et à niveaux de gris. Dans la deuxième partie, nous testerons la capacité des moments de Meixner à reconstituer des images binaires et en niveaux de gris avec et sans bruit.

5.1. Le temps de calcul des moments de Meixner

Nous allons comparer le temps de calcul des moments de Meixner selon trois méthodes: la méthode directe basée sur l'équation (III.20), la méthode proposée basée sur l'utilisation des blocs fixes (III.32) et l'application des algorithmes IBR-ISR, tels que définis précédemment, par (III.20) et (III.37).

Dans le premier exemple, un ensemble de cinq images binaires de dimensions 256x256 (III.10) sélectionnées dans la base de données bien connue MPEG-7 CE-shape-1 Part B [MPEG-7, 2014].

Le nombre de blocs de ces images, en utilisant la représentation par bloc, est NB = 292 pour Cheval, NB = 214 pour Bird, NB = 299 pour Elephant, NB = 220 pour Bat et NB = 47 pour Hammer (III.11).

Dans le deuxième exemple, un ensemble de cinq images en niveaux de gris avec une taille de 256x256, illustrée dans la figure (III.13), est utilisé. Le nombre de blocs et de tranches de ces images est: NB = 53701, NS = 216 pour Lena, NB = 55308; NS = 239 pour le bateau, NB = 61610; NS = 222 pour Mandrill, NB = 53410; NS = 238 pour le poivre et NB = 55879; NS = 218 pour Elaine (III.14).

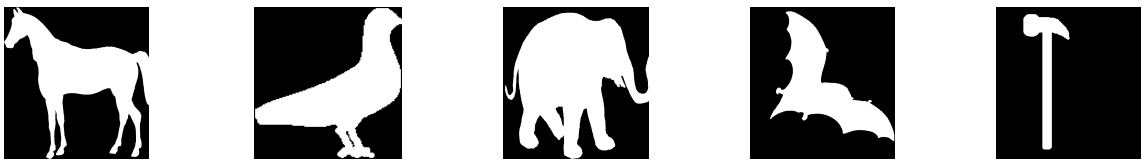
Le processus de calcul des moments orthogonaux discrets de Meixner est effectué à l'aide de:

La méthode slice-block (IBR-ISR): sachant que chaque bloc est une région rectangulaire, les moments sont calculés pour des ordres allant de 0 à la taille de chaque bloc pour chacune des cinq images.

la méthode de partitionnement de l'image en blocs fixes: chacune des cinq images contient 4096 blocs de taille [4x4] et 1024 blocs de taille [8x8] (III.12 et III.15). Les moments sont calculés pour des ordres allant de 0 à 4 pour des blocs de taille [4x4] et de 0 à 8 pour des blocs de taille [8x8].

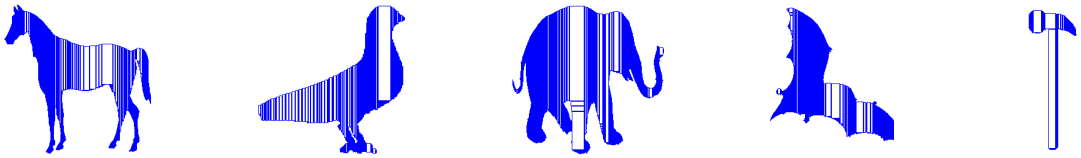
Les tableaux (III.1) et (III.2) représente le temps moyen de calcul des moments orthogonaux discrets de Meixner pour les cinq images binaires et les cinq images en niveaux de gris en utilisant les trois méthodes décrites précédemment.

Notez que l'algorithme a été implémenté sur un PC Dual Core 2.10 GHz, 2 Go de RAM.



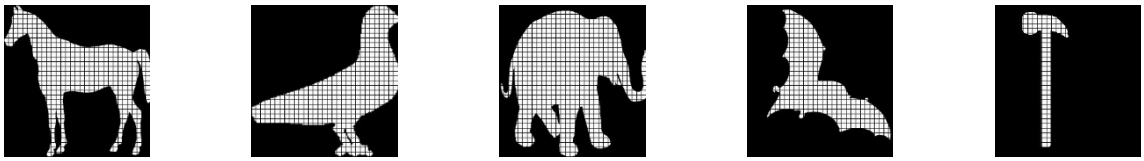
(a) (b) (c) (d) (e)

Figure (III.10): Ensemble d'images binaires de dimensions 256x256 (a) Horse, (b) Bird, (c) Elephant, (d) Bat and (e) Hammer



(a) (b) (c) (d) (e)

Figure (III.11): Représentation d'image par blocs (IBR): (a) Horse, (b) Bird, (c) Elephant, (d) Bat and (e) Hammer

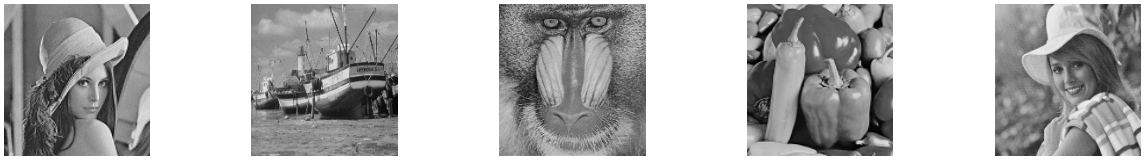


(a) (b) (c) (d) (e)

Figure (III.12): Partition d'image binaire en blocs fixes de tailles 8x8:Nb=256 (a) Horse, (b) Bird, (c) Elephant, (d) Bat and (e) Hammer

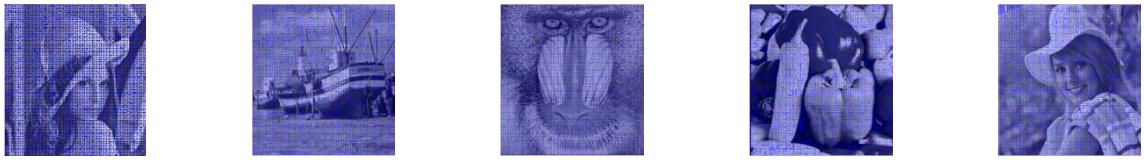
Tableau (III.1): Temps moyens pour le calcul des moments Meixner pour des images binaires

Méthode	Méthode directe	Blocs fixes [8x8]	Blocs fixes [4x4]	ISR-IBR
Temps moyen	0.0377	3.0431e-004	1.6822e-004	6.0065e-005



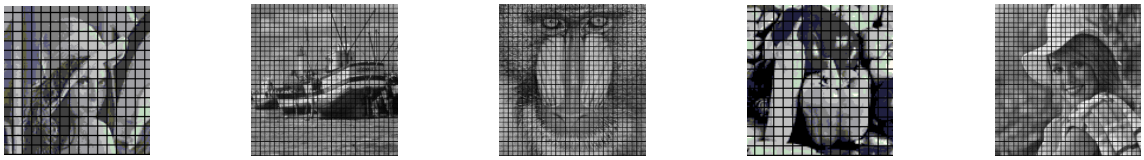
(a) (b) (c) (d) (e)

Figure (III.13): Ensemble d'image de niveaux de gris de domensions 256x256 (a) Lena, (b) Boat, (c) Mandrill, (d) Pepper and (e) Elaine



(a) (b) (c) (d) (e)

Figure (III.14): Représentation par slice-bloc (ISR-IBR): (a) Lena, (b) Boat, (c) Mandrill, (d) Pepper and (e) Elaine



(a) (b) (c) (d) (e)

Figure (III.15): partition en blocs fixes 8x8:Nb=256:(a) Lena, (b) Boat, (c) Mandrill, (d) Pepper and (e) Elaine

Tableau (III.2): Temps moyens de calcul des moments de Meixner pour des images de niveaux de gris

Method	Direct method	LBBRM size [8x8]	LBBRM size [4x4]	IBR-ISR
Average time (s)	0.0377	4.2109e-004	2.2873e-004	8.0874e-005

Les deux tableaux montrent que les méthodes proposées sont plus rapides que les méthodes directes, car le calcul des moments de Meixner par les deux méthodes proposées dépend uniquement du nombre de blocs au lieu de la taille de l'image.

5.2. La reconstruction d'image 2D par les méthodes proposées

Dans cette section, nous illustrons les performances des approches proposées. Cette dernière est testée sur deux images binaires de taille 256x256 pixels "Oiseau" et "éléphant" extraites de la base de données d'images MPEG-7 de la CE-shape-1, partie B [MPEG-7, 2014] et deux images en niveaux de gris de taille 256x256 "Barbara" et "Lena". Le test de reconstruction est effectué en utilisant les moments orthogonaux discrets de Meixner avec les deux méthodes. Nous allons calculer l'erreur au carré moyenne (MSE) définie par l'équation (III.25) entre l'original et l'image reconstruite.

La méthode de reconstruction par blocs fixes de taille [4x4] et [8x8]: ordre de reconstruction 4 pour la taille de blocs [4x4] et 8 pour la taille de blocs [8x8].

La méthode de reconstruction par IBR-ISR: L'ordre de reconstruction est égal à la taille du bloc. L'avantage de cette méthode est que seule l'intensité du premier et du dernier pixel de chaque bloc est calculée (III.16), l'intensité des autres pixels du bloc est estimée égale à la valeur moyenne des intensités des deux pixels reconstruits.

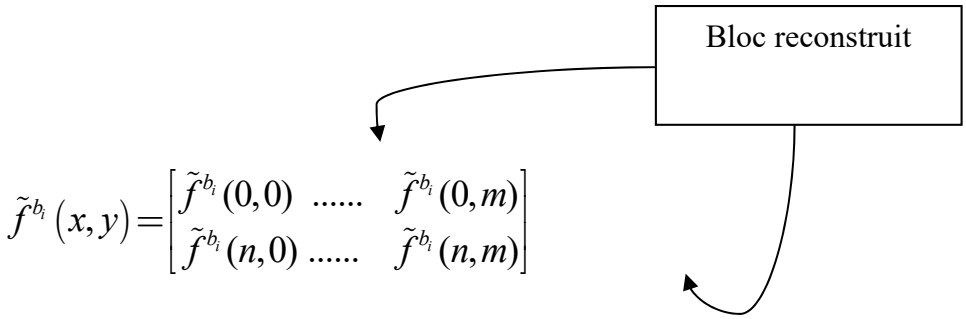


Figure (III.16): Reconstruction d'un bloc.

Les résultats de reconstruction des images de test utilisant les moments orthogonaux discrets de Meixner sont présentés dans le tableau (III.3). Ce dernier montre l'erreur quadratique moyenne (MSE) et les images reconstruites. De plus, les polynômes de Meixner dépendent des paramètres de localisation μ and β qui offrent un large choix dans la reconstruction.

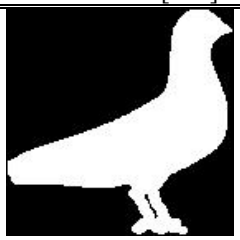
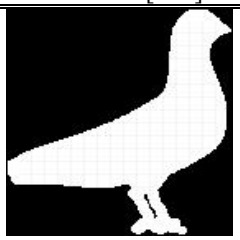
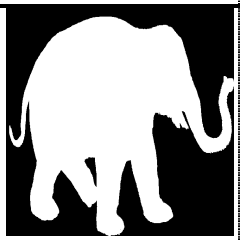
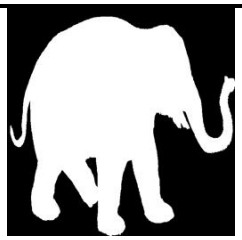
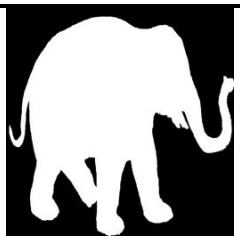
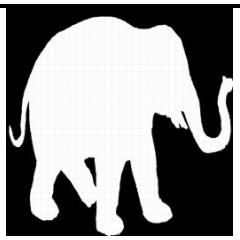
Le tableau montre que les erreurs de reconstruction par les deux méthodes proposées sont moins importantes. Cela montre également que les méthodes proposées sont efficaces en termes de qualité de reconstruction pour les images binaires et en niveaux de gris.

Dans la simulation suivante, nous avons testé la robustesse des moments orthogonaux discrets de Meixner par rapport aux différents types de bruit. Deux expériences sont donc effectuées sur une image en niveaux de gris "Barbara" de taille 256x256 (III.17.a) en utilisant deux types de bruit:

- The first type of noise is "salt and pepper" of density 0.03.
- The second type is “Gaussian noise” with zero mean and variance equal to 0.001.

Les images contaminées par les deux types de bruit sont présentées dans (III.17.b) et (III.17.c) et sont reconstruites à l'aide de moments orthogonaux discrets de Meixner. Le tableau (III.4) montre l'erreur quadratique moyenne (MSE) des images reconstruites.

Tableau (III.3): Images reconstruites et leurs Mean squared error (MSE)

Method	reconstruction by IBR-ISR	reconstruction method by LBBRM of size [4x4]	reconstruction method by LBBRM size [8x8]
 Bird	 *MSE=0.0095*	 *MSE=0.0278*	 *MSE= 0.2206*
 Elephant	 *MSE= 0.0012*	 *MSE= 0.0072*	 *MSE=0.5202*
 Lena	 MSE=0.0826	 MSE= 0.5185	 MSE= 0.7553
 Barbara	 MSE=0.0176	 MSE=0.4523	 MSE= 0.7429


(a)


(b)


(c)

Figure (III.17): (a) « Barbara » image de niveaux de gris de dimensions 256x256, (b) noisy by Salt & Peppers (bruité en sel et poivre) and (c) noisy image by white Gaussian (bruit Gaussian blanc)

Tableau (III.4): Images reconstruites et leurs Mean squared error (MSE).

Method	reconstruction by IBR-ISR	reconstruction method by LBBRM of size [4x4]	reconstruction method by LBBRM of size [8x8]
Selt & poivre Noise	 MSE=0.7744	 MSE=0.7744	 MSE=0.7114
Gaussian Noise	 MSE=0.0201	 MSE=0.4413	 MSE= 0.7414

Nous concluons que les erreurs quadratiques moyennes des deux images bruitées sont supérieures à celles de l'image sans bruit (ci-dessus) mais elles sont très petites, ce qui montre la robustesse des méthodes proposées basées sur les moments de Meixner par rapport aux deux types de bruit.

6. Conclusion

Dans ce chapitre, nous avons proposé une méthode de calcul des moments orthogonaux discrets de Meixner. L'Approche directe présente l'instabilité numérique des moments d'ordre élevé. Pour résoudre ce problème, nous avons adopté deux algorithmes basés sur le partitionnement de l'image en un ensemble de petites régions. Ces dernières sont purement locales, chaque bloc étant traité indépendamment. Ainsi, la réduction de l'espace d'image permet d'utiliser des moments orthogonaux discrets de faibles ordres dans le processus de calcul des moments de Meixner orthogonaux discrets. Ainsi, la reconstruction de l'image par ces méthodes entraîne une réduction considérable de l'erreur de reconstruction et une robustesse contre les bruits qui deviennent locaux. Les résultats importants des simulations confirment que la méthode proposée sera efficace dans le futur, principalement dans la reconstruction d'images, mais également dans la classification et l'indexation des images afin de proposer une nouvelle méthode de recherche par contenu dans des bases d'images spécialisées ou générales.

CHAPITRE 4 : ALGORITHME RAPIDE DE CALCUL DES MOMENTS ORTHOGONAUX DISCRETS SUR DES IMAGES 3D BASE SUR LA ICR

L'essor de l'imagerie numérique est remarquable, les méthodes et techniques de traitement et d'analyse de l'image numérique doivent également accompagner cette évolution technologique. Dans un axe de recherche sur la théorie des moments associés à l'imagerie numérique, les valeurs sont extraites des images numériques pour les besoins de classifications ou même de reconstruction, en tant que descripteurs uniques d'une image. Dans ce chapitre, nous proposons une nouvelle méthode rapide et efficace pour calculer des moments orthogonaux sur une image 3D discrète. Nous avons opté pour les polynômes orthogonaux de Meixner et pour une nouvelle représentation de l'image 3D par des cuboïdes ayant des niveaux de gris homogènes, appelés représentation d'image par les cuboïdes (ICR). Sur la base de cette représentation, nous calculons les moments sur chaque cuboïde avant de les additionner pour obtenir les moments globaux d'une image 3D. A travers un ensemble de simulations, nous prouvons que notre méthode permet de réduire le temps nécessaire au calcul du moment sur une image 3D de toute taille et pour tout ordre, mais pas seulement, cette méthode permet d'améliorer la qualité de l'image 3D reconstruite à partir des moments de petits ordres.

1. Introduction

Plusieurs algorithmes ont été proposés pour réduire le temps de calcul des moments en réduisant le nombre d'opérations nécessaires pour obtenir les valeurs des moments de différents ordres [Hornavr, 2013; Shu, 2010; Hwang, 2006; Spiliotis, 1998]. Parmi celles-ci, nous retrouvons la méthode récursive par rapport à la variable n et à la variable x, à l'utilisation de la formule de Clenshaw et au calcul matriciel [Wang, 2006; Fu, 2008]. La plupart ou la totalité des solutions proposées, centrées uniquement sur les images 2D, les images 3D n'étaient pas suffisamment étudiées, tandis que le calcul des moments sur des images 3D représente un défi majeur. La taille de cette dernière, même en appliquant les

algorithmes proposés pour les images 2D sur des images 3D, nous remarquons que le temps de calcul augmente considérablement, ce qui pose un réel problème.

Pour surmonter ce problème, nous avons considéré le problème du calcul des moments sur une image 3D comme un ensemble de problèmes. Au fait, nous pouvons subdiviser l'image en plusieurs blocs 3D en calculant le moment sur chaque bloc séparément, puis en rassemblant les moments obtenus pour obtenir les moments globaux sur l'ensemble de l'image, nous pourrons peut-être réduire le nombre énorme de produits et additions pouvant impliquer le calcul de moments directement à partir d'une image 3D, cela résoudra également le problème de divergence des résultats lorsqu'il s'agit d'un ordre de calcul plus grand.

Ainsi, nous nous sommes inspirés des travaux de Spiliotis et Papakostas [Spiliotis, 1998; Papakostas, 2008], qui subdivisaient les images 2D en tranches et en blocs avant d'effectuer les calculs des moments. Ainsi, nous imaginons le volume 3D ou l'image 3D comme un ensemble de blocs 3D que nous avons appelés cuboïdes (le terme cuboïde désignant un bloc 3D d'une largeur, d'une longueur et d'une hauteur), ils ne contiennent que des voxels de même intensité. Nous obtiendrons un cuboïde homogène avec une certaine intensité, ce qui permettra d'éliminer la lecture de la valeur de chaque voxel du cuboïde avant de calculer les moments, ainsi que l'intensité d'un voxel est l'intensité du cuboïde. En conséquence, nous pouvons encore améliorer la vitesse de calcul en introduisant le calcul matriciel dans le calcul des moments sur chaque cuboïde pour augmenter la rapidité de notre algorithme.

Le choix des polynômes de Meixner n'est pas arbitraire, ces derniers n'ont pas été suffisamment étudiés pour le calcul des moments, encore moins par rapport aux polynômes de Tchebichef, Krawtchouk, Hahn ou Charlier.

Ce chapitre est organisé comme suit: dans la première section, nous définissons les moments orthogonaux discrets (qui seront utilisés dans les simulations). Dans la section suivante, nous présenterons une nouvelle représentation d'image 3D basée sur la représentation d'image par cuboïdes (ICR). Dans la troisième partie, nous présentons un nouvel algorithme pour le calcul rapide des moments 3D de Meixner, basé sur la représentation d'images 3D par des cuboïdes et le produit matriciel. Nous présentons

également la nouvelle méthode de reconstruction d'images 3D basée sur la reconstruction d'images à partir de cuboïdes extraits par ICR. Dans la dernière section, nous présentons les résultats de la simulation afin de prouver l'efficacité et la rapidité de l'algorithme proposé.

2. Moments et polynômes orthogonaux discrets : forme générale

L'exploitation des propriétés intéressantes de certains polynômes orthogonaux est devenue de plus en plus liée à l'extraction de caractéristiques d'image. Au fil de nombreuses études, les chercheurs ont constamment amélioré l'utilisation des polynômes orthogonaux dans la reconstruction et la classification des images à l'aide de calculs de moments et de leurs inverses.

Les polynômes orthogonaux discrets sont définis comme les solutions de polynômes de l'équation de différence suivante [Nikiforov, 1991]:

$$\sigma(x)\Delta\nabla P_n(x)+\tau(x)\Delta P_n(x)+\lambda_n P_n(x)=0 \qquad \textbf{(IV.1)}$$

Où

$$\Delta P_n(x)=P_n(x+1)-P_n(x)\quad \nabla P_n(x)=P_n(x)-P_n(x-1)\quad \Delta P_n(x)=\nabla P_n(x+1)$$

Considérant les propriétés de l'opérateur ∇ et Δ [36], nous avons:

$$\Delta\nabla P_n(x)=\nabla\Delta P_n(x)=P_n(x+1)-2P_n(x)+P_n(x-1) \qquad \textbf{(IV.2)}$$

Désigne l'opérateur de différence finie forward et backward, respectivement.

$\sigma(x)$ et $\tau(x)$ sont les fonctions de deuxième et premier degré, respectivement. λ_n est une constante appropriée. La solution de cette équation aux différences partielles peut être exprimée par la formule de Rodrigues comme suit:

$$P_n(x)=\frac{B_n}{\omega(x)}\nabla^n\left[\omega_n(x)\right] \qquad \textbf{(IV.3)}$$

où $\omega(x)$ est la fonction de poids dans le cas $n=0$. Ainsi, les solutions de polynômes de (IV.1) sont déterminées par (IV.3) en fonction des facteurs de normalisation B_n

. Pour l'opérateur de différence ‘forward’ ∇, nous avons la propriété [Wee, 2007; Zhang, 2010]:

$$\nabla^n f(x) = \sum_{k=0}^{n} \binom{n}{k} (-1)^k f(x-k) \tag{IV.4}$$

En combinant la formule de Rodrigues et (IV.4), nous pouvons obtenir une expression explicite pour les polynômes $P_n(x)$.

Les polynômes $P_n(x)$ satisfont une relation d'orthogonalité de la forme:

$$\sum_{x=0}^{s} P_n(x) P_m(x) \omega(x) = \rho_n \delta_{nm}, \qquad 0 \le n, m \le s \tag{IV.5}$$

Ainsi, les relations de récurrence de polynômes orthogonaux discrets avec respect à x peuvent être obtenues par (IV.1) et (IV.2) comme suit:

$$P_n(x) = \frac{2\sigma(x-1) + \tau(x-1) - \lambda_n}{\sigma(x-1) + \tau(x-1)} P_n(x-1) - \frac{\sigma(x-1)}{\sigma(x-1) + \tau(x-1)} P_n(x-2) \tag{IV.6}$$

Les polynômes orthogonaux normalisés peuvent être obtenus en utilisant la norme carrée $\rho(n)$ et la fonction du poids $\omega(x)$:

$$\tilde{P}_n(x) = P_n(x) \sqrt{\frac{\omega(x)}{\rho(n)}} \tag{IV.7}$$

Par conséquent, la propriété orthogonale des polynômes orthogonaux normalisés dans (IV.3) peut être réécrite comme suit:

$$\sum_{x=0}^{s} \tilde{P}_n(x) \tilde{P}_m(x) = \delta_{nm}, \qquad 0 \le n, m \le s \tag{IV.8}$$

Selon (IV.6) et (IV.7), une autre forme générale d'obtention de polynômes orthogonaux discrets est écrite comme suit. C'est une relation de récurrence par respect à x:

$$\tilde{P}_n(x)=\frac{2\sigma(x-1)+\tau(x-1)-\lambda_n}{\sigma(x-1)+\tau(x-1)}\sqrt{\frac{\omega(x)}{\omega(x-1)}}\tilde{P}_n(x-1)-\frac{\sigma(x-1)}{\sigma(x-1)+\tau(x-1)}\sqrt{\frac{\omega(x)}{\omega(x-2)}}\tilde{P}_n(x-2)$$

(IV.9)

Ensuite, les relations de récurrence de polynômes orthonormaux discrets de Tchebichef, Krawtchouk, et Meixner par rapport à la variable sont obtenues à partir de la forme générale Eq. (IV.9) et les données du tableau (IV.1).

Tableau (IV.1) : Polynômes Orthogonal généralisés

$P_n(x)$	$T_n(x)$	$K_n^p(x)$ $0\prec p\prec 1$	$H_n^{(a,b)}(x)$ $a\succ 0,\ b\succ 0$	$C_n^{a_1}(x)$ $a_1\succ 0$	$M_n^{(\gamma,\mu)}(x)$ $\gamma\succ 0,\ 0\prec\mu\prec 1$
s	$N-1$	N	$N-1$	∞	∞
$\sigma(x)$	$x(N-x)$	$x(1-p)$	$x(N+a-x)$	x	x
$\tau(x)$	$N-1-2x$	$(Np-x)$	$(b+1)(N-1)-(a+b+2)x$	a_1-x	$\gamma\mu-x(1-\mu)$
λ_n	$n(n+1)$	n	$n(n+1+a+b)$	n	$n(1-\mu)$
$\omega(x)$	1	$\binom{N}{x}p^x(1-p)^{N-x}$	$\frac{\Gamma(N+a-x)\Gamma(b+x+1)}{\Gamma(N-x)\Gamma(x+1)}$	$\frac{e^{-a_1}a_1^x}{x!}$	$\frac{\mu^x\Gamma(x+\gamma)}{\Gamma(x+1)\Gamma(\gamma)}$
$\rho(n)$	$\frac{\Gamma(N+n+1)}{(2n+1)\Gamma(N-n)}$	$(-1)^n\left(\frac{1-p}{p}\right)^n\frac{n!}{(-N)_n}$	$\frac{\Gamma(a+n+1)\Gamma(b+n+1)(a+b+n+1)_N}{(a+b+2n+1)(N-n-1)!n!}$	$\frac{n!}{a_1^n}$	$\frac{\Gamma(n+1)(\gamma)_n}{\mu^n(1-\mu)^\gamma}$

Tableau (IV.2):les termes initiaux

$\tilde{P}_n(x)$	$\tilde{P}_n(x=0)$	$\tilde{P}_n(x=1)$
$\tilde{T}_n(x)$	$(-1)^n(N-n)_n\sqrt{\frac{\omega(0)}{\rho(n)}}$	$\left(1+\frac{n(n+1)}{1-N}\right)\sqrt{\frac{\omega(1)}{\omega(0)}}\tilde{T}_n(0)$
$\tilde{K}_n^p(x)$	$\sqrt{\frac{\omega(0)}{\rho(n)}}$	$\left(1-\frac{n}{Np}\right)\sqrt{\frac{\omega(1)}{\rho(n)}}$
$\tilde{H}_n^{(a,b)}(x)$	$(1-N)_n\binom{n+b}{b}_n\sqrt{\frac{\omega(0)}{\rho(n)}}$	$\frac{(n+b+1)(N-n-1)-n(N+a-1)}{(b+1)(N-1)}\sqrt{\frac{\omega(1)}{\omega(0)}}\tilde{H}_n^{(a,b)}(0)$
$C_n^{a_1}(x)$	$\sqrt{\frac{\omega(0)}{\rho(n)}}$	$\frac{a_1-n}{a_1}\sqrt{\frac{\omega(1)}{\rho(n)}}\tilde{C}_n^{a_1}(0)$
$\tilde{M}_n^{(\gamma,\mu)}(x)$	$(\gamma)_n\sqrt{\frac{\omega(0)}{\rho(n)}}$	$\frac{\mu(n+\gamma)-n}{\mu\gamma}\sqrt{\frac{\omega(1)}{\omega(0)}}\tilde{M}_n^{(\gamma,\mu)}(x)$

En utilisant de la formule de Rodriguez (IV.3), et la relation d'orthogonalité (IV.5) et le tableau (IV.1), nous obtenons les valeurs initiales de la relation de récurrence par rapport à la variable x, comme indiqué dans le tableau (IV.2). Les équations ci-dessus peuvent être utilisées pour calculer les polynômes pondérés discrets de Tchebichef, Krawtchouk, Hahn, Charlier et Meixner, respectivement. A noter que la même méthode peut être appliqué sur tout les autres polynômes orthoganaux.

L'algorithme suivant est utilisé pour calculer les valuers des polynômes orthogonaux discrtes par la récurssive en trois termes en respect à x.

Input: Order, N
Output*:* Polynomial matrix $\tilde{P}_n(x)$
Compute norm, $\rho(n)$ from Table 1;
Compute weight, $\omega(x)$ from Table 1;
for $x \leftarrow 0$ *to* N-1 **do**
if $(x=0)$
for $n \leftarrow 0$ *to order* **do**
Compute $\tilde{P}_n(x=0)$ from Table 2;
end
elseif $(x=1)$
for $n \leftarrow 0$ *to order* **do**
Compute $\tilde{P}_n(x=1)$ from Table 2;
end
else
for $n \leftarrow 0$ *to order* **do**
Compute $\tilde{P}_n(x)$ from Eq. (9);
end
end
end

Plus loin dans ce chapitre, nous utiliserons les polynômes de Meixner pour calculer les moments et leurs inverses avec une nouvelle méthode plus rapide et plus efficace par rapport à la méthode récursive de *x*, le but étant d'introduire le calcul des moments de Meixner et leurs inverses pour les images en 3 dimensions.

3. Représentation d'image 3D par cuboïdes (ICR)

Dans cette section, nous proposons une nouvelle méthode de représentation des images 3D par des cuboïdes contenant des voxels de même intensité (homogènes). Grâce à cette représentation, l'image sera définie par un ensemble de cuboïdes où chaque cuboïde contient un seul niveau de gris. Les images ou volumes 3D sont des représentations en trois dimensions d'un objet ou d'un ensemble d'objets du domaine réel dans le domaine numérique. Dans le cas de représentations discrètes, ces images ou volumes 3D sont souvent projetés dans un espace tridimensionnel sur les trois axes des x,y et z .

La notion de voxel dans les images 3D est l'équivalent du pixel dans les images 2D, le voxel représente donc l'unité élémentaire du volume, qui est définie par sa position dans l'espace, ses dimensions et son intensité. Par conséquent, les voxels sont souvent considérés comme des éléments ayant les mêmes dimensions, mais il peut arriver que les voxels de la même image aient des dimensions différentes, comme dans les images médicales IRM. De plus, dans le cas de voxels de volume, ayant les mêmes dimensions, on considère que le volume peut être représenté par une fonction discrète $f(x,y,z)$ où chaque voxel de coordonnées (x,y,z) est représenté par une intensité f_i, ainsi on peut représenter cette fonction sous forme de matrice 3D.

La représentation ICR permet de diviser l'image en éléments plus importants qu'un simple voxel, d'intensité homogène, nous aurons donc moins d'éléments à traiter. Avec cette nouvelle représentation, il ne sera plus nécessaire de décrire les voxels de niveau *'0',* mais uniquement les voxels de niveau *'1'* pour le cas des images binaires 3D. Dans le cas des images en niveaux de gris, l'intensité prend des valeurs comprises entre 0 et 255, cela signifie que nous aurons des cuboïdes représentés par $(1 \le n \le 255)$ n valeurs d'intensité. Les cuboïdes d'intensité nulle seront négligés. Ainsi, pour représenter des images 3D de taille $N \times M \times K$ binaire ou en niveaux de gris, nous proposons l'algorithme suivant appelé représentation cuboïde (ICR).

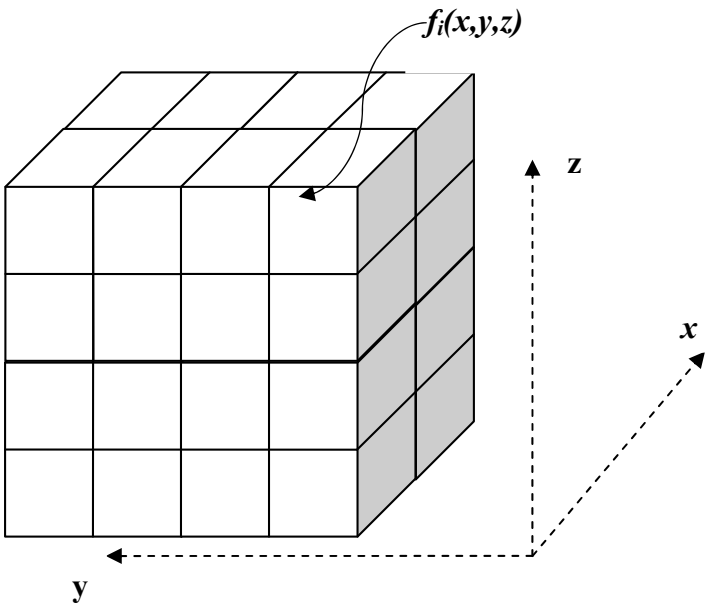


Figure (IV.1): Représentation d'image 3D discrète dans l'espace

Algorithm - ICR

Etape 1: Diviser l'image $f(x,y,z)$ tel que $x \in [1,N]$, $y \in [1,M]$ *and* $z \in [1,K]$ sur K matrice de dimensions $N \times M$.

Etape 2: Déterminer les blocs partiels sur chaque matrice K_s avec $s = 0,... K-1$

Etape 3: les comparer aux blocs de mêmes intensités dans la matrices K_{s-1}. Si aucun bloc ne correspond, créer un nouveau cuboïde.

Etape 4: En sortie, nous aurons L cuboïdes décrits par leurs intensités f_i et leurs coordonnées $(x_1, x_2, y_1, y_2, z_1, z_2)$.

Afin d'expliquer plus en détail l'algorithme proposé, nous l'appliquerons à une image 3D tirée de la base de données de Mcgill [Mcgill, 2017], qui représente un "Ant"(fourmis) en 3D de taille $128 \times 128 \times 128$ (figure IV.2). En appliquant l'algorithme ICR sur cette image, nous obtenons les différents cuboïdes qui composent l'image, comme le montre la figure (IV.3). Le nombre de cuboïdes dans cette image est de 586 cuboïdes au lieu de $128 \times 128 \times 128 = 2\,097\,152$ voxels. Ce qui montre l'importance de cette nouvelle représentation pour le traitement de l'image.

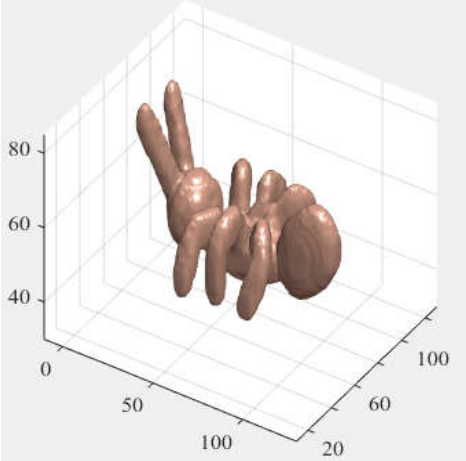


Figure (IV.2): Image "Ant" de la base McGill

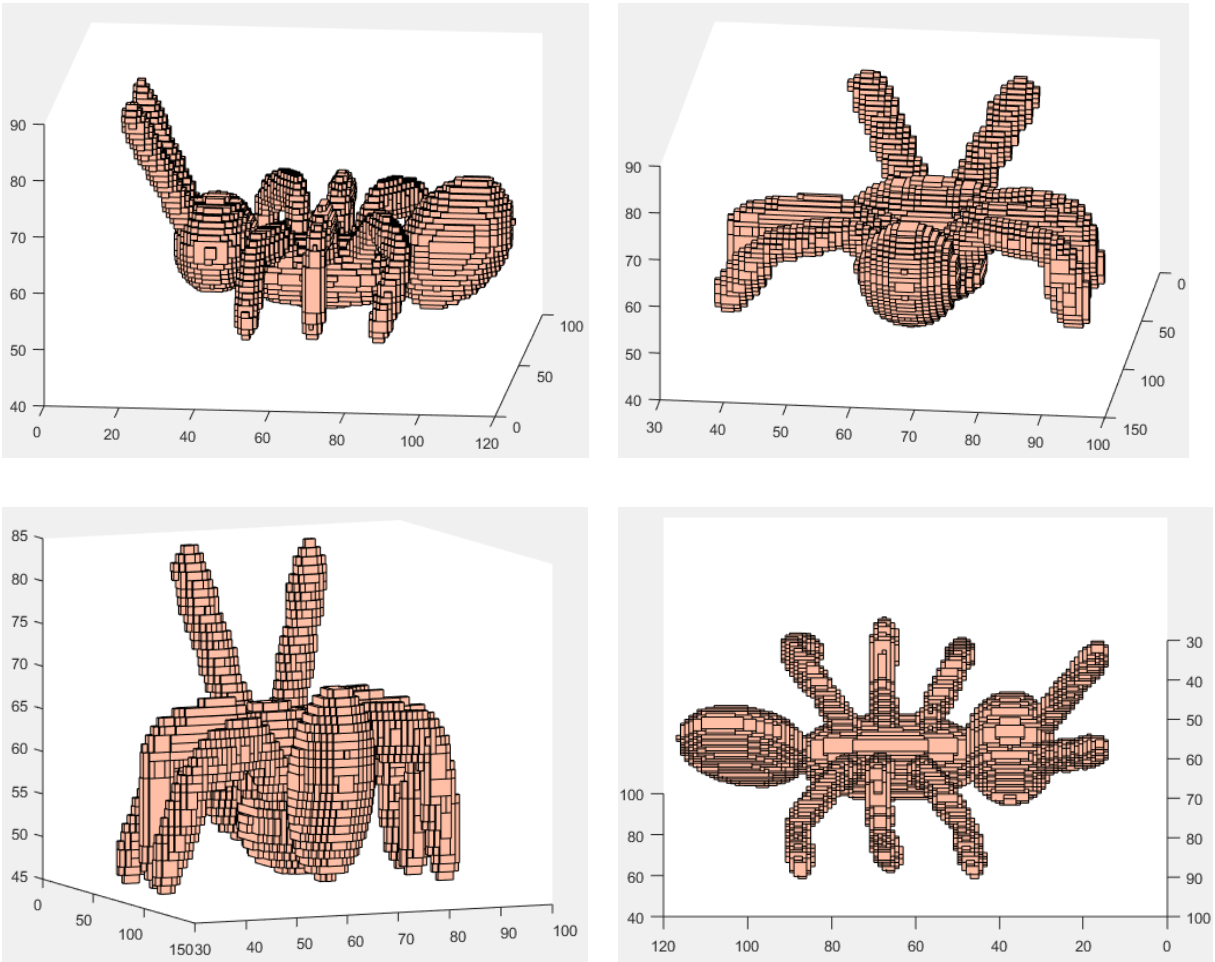


Figure (IV.3): plusieurs angles de vues de la représentation de l'image "Ant" par cuboïdes.

Par conséquent, la méthode proposée permet de diviser un volume 3D en plusieurs cuboïdes homogènes de tailles différentes pour tous les niveaux de gris inclus dans le volume, ce qui permettra de calculer les moments de Meixner et leurs inverses séparément sur chaque cuboïde, ce qui implique nécessairement la réduction des nombres des opérations de produits et des additions nécessaires pour obtenir les résultats convoités, à partir des moments calculés sur chaque cuboïde, nous pourrons déduire les moments de Meixner et leurs inverses sur tout le volume 3D, et gagner ainsi en rapidité et en efficacité.

4. Calcul rapide des moments de Meixner et leurs inverses via l'ICR

Dans cette section, nous exposerons notre méthode pour calculer les moments de Meixner et leurs inverses à l'aide des cuboïdes extraits d'un volume discret 3D. La méthode que nous proposons implique deux approches, l'approche directe qui calcule les moments et leurs inverses sur chaque cuboïde en utilisant la méthode récursive des polynômes de Meixner par rapport à la variable x, et l'approche matricielle qui permet de les calculer via le produit matriciel.

4.1. Calcul des moments de Meixners et leurs inverses : méthode classique

Les moments de Meixner en trois dimensions 3D sont définis comme les projections d'une image 3D sur des polynômes orthogonaux de Meixner. Par conséquent, pour une image discrète 3D $f(x,y,z)$ de taille $N\times M\times K$, $x\in[0,\ N-1]$, $y\in[0,\ M-1]$ *and* $z\in[0,\ K-1]$, l'ordre $(n+m+k)^{th}$ des moments de Meixner est défini comme suit:

$$MM_{nmk}=\sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\sum_{z=0}^{K-1}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z)f(x,y,z) \qquad \textbf{(IV.10)}$$

où $\tilde{M}_n^{(\gamma,\mu)}(x)$ est le n^{th} polynôme orthogonal normalisé de Meixner.

Par le principe d'orthogonalité, l'image 3D f(x,y,z) peut être représenté par une série de sommes en terme des polynômes orthogonaux discrets, ainsi l'équation (IV.10) permettra d'exprimer la transformée inverse comme suit :

$$\tilde{f}(x,y,z)=\sum_{n=0}^{\infty}\sum_{m=0}^{\infty}\sum_{k=0}^{\infty}MM_{nmk}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z) \quad \textbf{(IV.11)}$$

4.2. Calcul rapide des moments de Meixner et leurs inverses : méthode ICR

Cette première approche est basée sur les cuboïdes extraits sur le volume discret 3D en utilisant la méthode ICR proposée précédemment, qui permet de présenter une image discrète 3D $f(x,y,z)$ pat L cuboïdes comme suit:

$$f(x,y,z)=\begin{cases}\sum_{i=1}^{255} f_i(x,y,z)\\ f_i:\ The\ i^{th}\ \text{int}ensity\ of\ \ 3D\ image,\ 1\le i\le 255\end{cases} \quad \textbf{(IV.12)}$$

Où

$$f_i(x,y,z)=\begin{cases}Cub_j^{f_i}\ \ , j=1,\cdots,L_i,\ where\ Cub_j^{f_i}\ is\ the\ cuboid\ j\ of\ \text{int}ensity\ f_i\\ \qquad\quad , j\ is\ the\ j^{th}\ cuboid\ of\ \text{int}ensity\ f_i\end{cases} \quad \textbf{(IV.13)}$$

avec

$$L=\sum_{i=1}^{255}L_i \quad , L_i\ the\ number\ of\ cuboid\ for\ each\ \text{int}ensity\ f_i$$

En utilisant l'algorithme ICR, les moments 3D de Meixner d'ordre $(n+m+k)^{th}$ de l'image 3D $f(x,y,z)$ de dimension $N\times M\times K$ dans l'équation (IV.10), peut être calculé comme suit:

$$
\begin{aligned}
MM_{nmk} &= \sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\sum_{z=0}^{K-1}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z)f(x,y,z) \\
&= \sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\sum_{z=0}^{K-1}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z)\sum_{i=1}^{255}f_i(x,y,z) \\
&= \sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\sum_{z=0}^{K-1}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z)f_1(x,y,z) \\
&+ \sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\sum_{z=0}^{K-1}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z)f_2(x,y,z)+\cdots\cdots \\
&+ \sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\sum_{z=0}^{K-1}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z)f_{255}(x,y,z)
\end{aligned}
\qquad \textbf{(IV.14)}
$$

Les cuboïdes sont déterminés par les coordonnées $(x_1, x_2, y_1, y_2, z_1, z_2)$, la formule précédente dans l'équation (14) est réécrite comme suit:

$$
\begin{aligned}
MM_{nmk} &= f_1\sum_{x_1}\sum_{y_1}\sum_{z_1}\tilde{M}_n^{(\gamma,\mu)}(x_1)\tilde{M}_m^{(\gamma,\mu)}(y_1)\tilde{M}_k^{(\gamma,\mu)}(z_1)+f_2\sum_{x_2}\sum_{y_2}\sum_{z_2}\tilde{M}_n^{(\gamma,\mu)}(x_2)\tilde{M}_m^{(\gamma,\mu)}(y_2)\tilde{M}_k^{(\gamma,\mu)}(z_2) \\
&+\cdots f_{255}\sum_{x_{255}}\sum_{y_{255}}\sum_{z_{255}}\tilde{M}_n^{(\gamma,\mu)}(x_{255})\tilde{M}_m^{(\gamma,\mu)}(y_{255})\tilde{M}_k^{(\gamma,\mu)}(z_{255}) \\
&= f_1 MM_{nmk}^1 + f_2 MM_{nmk}^2 + \cdots f_{255} MM_{nmk}^{255} \\
&= \sum_{i=1}^{255} f_i MM_{nmk}^i
\end{aligned}
\qquad \textbf{(IV.15)}
$$

où MM_{nmk}^{i} est le moment 3D de Meixner d'ordre $(n+m+k)^{th}$ de la function d'intensité f_i.

Les moments correspondant à l'intensité f_i sont calculés à partir des cuboïdes de chaque intensité par la relation suivante:

$$
\begin{aligned}
MM_{nmk}^{i} &= \sum_{j=1}^{L_i} f_i \sum_{x=x_{1,Cub_j^{fi}}}^{x_{2,Cub_j^{fi}}} \tilde{M}_n^{(\gamma,\mu)}(x) \sum_{y=y_{1,Cub_j^{fi}}}^{y_{2,Cub_j^{fi}}} \tilde{M}_m^{(\gamma,\mu)}(y) \sum_{z=z_{1,Cub_j^{fi}}}^{z_{2,Cub_j^{fi}}} \tilde{M}_k^{(\gamma,\mu)}(z) \\
&= \sum_{j=1}^{L_i} MM_{nmk}^{Cub_j^{fi}}
\end{aligned}
\qquad \textbf{(IV.16)}
$$

où $MM_{nmk}^{Cub_j^{f_i}}$ est le moment 3D de Meixner d'ordre $(n+m+k)^{th}$ de la function d'intensité f_i pour chaque cuboide.

Ainsi, les moments orthogonaux discrets de Meixner 3D, peuvent être écrits comme suit:

$$\begin{aligned} MM_{nmk} &= \sum_{i=1}^{255}\sum_{j=1}^{L_i}\sum_{x=x_{1,Cub_j^{f_i}}}^{x_{2,Cub_j^{f_i}}}\sum_{y=y_{1,Cub_j^{f_i}}}^{y_{2,Cub_j^{f_i}}}\sum_{z=z_{1,Cub_j^{f_i}}}^{z_{2,Cub_j^{f_i}}}\tilde{M}_m^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z)f_i(x,y,z) \\ &= \sum_{i=1}^{255}\sum_{j=1}^{L_i} MM_{nmk}^{Cub_j^{f_i}} \end{aligned}$$

(IV.17)

Avec

$$MM_{nmk}^{Cub_j^{f_i}} = \sum_{x=x_{1,Cub_j^{f_i}}}^{x_{2,Cub_j^{f_i}}}\sum_{y=y_{1,Cub_j^{f_i}}}^{y_{2,Cub_j^{f_i}}}\sum_{z=z_{1,Cub_j^{f_i}}}^{z_{2,Cub_j^{f_i}}}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z)f_i(x,y,z)$$

(IV.18)

En théorie, une fonction originale de l'image 3D $f(x,y,z)$ de dimension $N\times M\times K$ peut être représentée à partir d'une série infinie de moments Meixner.

$$f(x,y,z) = \sum_{n=0}^{\infty}\sum_{m=0}^{\infty}\sum_{k=0}^{\infty} MM_{nmk}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z) \quad , \begin{cases} 0\le x\le N-1 \\ 0\le y\le M-1 \\ 0\le z\le K-1 \end{cases}$$

(IV.19)

La transformée inverse locale de Meixner Moments d'une image 3D $f(x,y,z)$ de taille $N\times M\times K$ par le calcul direct est obtenu à partir de la représentation par cuboïde d'image 3D (ICR), en utilisant la définition de moments de Meixner dans l'équation (IV.17):

$$f(x,y,z)=\sum_{i=1}^{255}\sum_{j=1}^{L_i}\sum_{n=0}^{\infty}\sum_{m=0}^{\infty}\sum_{k=0}^{\infty}MM_{nmk}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z) \quad ,\begin{cases} x_{1,Cub_j^{fi}} \le x \le x_{2,Cub_j^{fi}} \\ y_{1,Cub_j^{fi}} \le y \le y_{2,Cub_j^{fi}} \\ z_{1,Cub_j^{fi}} \le z \le z_{2,Cub_j^{fi}} \end{cases}$$

$$=\sum_{i=1}^{255}\sum_{j=1}^{L_i} f^{Cub_j^{fi}}(x,y,z)$$

(IV.20)

où $f^{Cub_j^{fi}}(x,y,z)$ est la reconstruction partielle pour chaque cuboïde:

$$f^{Cub_j^{fi}}(x,y,z)=\sum_{n=0}^{\infty}\sum_{m=0}^{\infty}\sum_{k=0}^{\infty}MM_{nmk}\tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z) \quad ,\begin{cases} x_{1,Cub_j^{fi}} \le x \le x_{2,Cub_j^{fi}} \\ y_{1,Cub_j^{fi}} \le y \le y_{2,Cub_j^{fi}} \\ z_{1,Cub_j^{fi}} \le z \le z_{2,Cub_j^{fi}} \end{cases}$$

(IV.21)

4.3. Calcul rapide des moments de Meixner et leurs inverses : Méthode matricielle

La seconde approche est proposée pour accélérer le calcul des moments de Meixner et de leurs inverses par le produit matriciel. Nous avons combiné la représentation de l'algorithme image cuboïde (ICR) avec l'algorithme basé sur le produit matriciel proposé par Wu et al [Wu, 2013]. Après avoir spécifié les cuboïdes d'une image 3D en utilisant l'algorithme (ICR), nous avons ensuite appliqué le calcul matriciel pour le calcul des moments de Meixner pour chaque matrice du cuboïde j de taille $N_j = x_2 - x_1 + 1, M_j = y_2 - y_1 + 1$ $and\ K_j = z_2 - z_1 + 1$ par les coordonnées $(x_1, x_2, y_1, y_2, z_1, z_2)$ basé sur la décomposition de ce cuboïde j dans K_j matrices, qui représente les blocs b_{ij}^z, $z_1 \le z \le z_2$, puis nous calculons les moments de Meixner dans chaque bloc à l'étape suivante. Ainsi, il est évident que chaque bloc est déterminé par les coordonnées (x_1, x_2, y_1, y_2).

L'expression des moments orthogonaux 3D de Meixner pour chaque cuboïde de l'équation (VI.18) peut être écrite comme suit:

$$MM_{nmk}^{Cub_j^{fi}} = \sum_{z=z_{1,Cub_j^{fi}}}^{z_{2,Cub_j^{fi}}} \tilde{M}_k^{(\gamma,\mu)}(z)\left[\sum_{x=x_{1,Cub_j^{fi}}}^{x_{2,Cub_j^{fi}}}\sum_{y=y_{1,Cub_j^{fi}}}^{y_{2,Cub_j^{fi}}} \tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y) f_i(x,y,z)\right]$$
$$= \sum_{z=z_{1,Cub_j^{fi}}}^{z_{2,Cub_j^{fi}}} \tilde{M}_k^{(\gamma,\mu)}(z) MM_{nm}^{b_{ij}^z}$$

(IV.22)

avec $MM_{nm}^{b_{ij}^z}$ les moments de Meixner dans chaque bloc de cuboïde j, peut être écrit également comme suit:

$$MM_{nm}^{b_{ij}^z} = \sum_{x=x_{1,Cub_j^{fi}}}^{x_{2,Cub_j^{fi}}}\sum_{y=y_{1,Cub_j^{fi}}}^{y_{2,Cub_j^{fi}}} \tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y) f_i(x,y,z) \qquad \textbf{(IV.23)}$$

qui peut être ré-exprimé en tant que produit matriciel:

$$MM_{nm}^{b_{ij}^z} = P_1 I P_2^T \qquad \textbf{(IV.24)}$$

$(.)^T$ désigne la transposée de la matrice, où:

$MM^{b_{ij}^z} = \left\{MM_{nm}^{b_{ij}^z}\right\}_{n,m=0}^{n=M-1,m=N-1}$, $0 \le n \le M-1$, $0 \le m \le N-1$

$P_1 = \left\{\tilde{M}_n^{(\gamma,\mu)}(x)\right\}_{n=0,x=x_{1,Cub_j^{fi}}}^{n=M-1,x=x_{2,Cub_j^{fi}}}$, $0 \le n \le M-1$, $x_{1,Cub_j^{fi}} \le x \le x_{2,Cub_j^{fi}}$

$P_2 = \left\{\tilde{M}_m^{(\gamma,\mu)}(y)\right\}_{m=0,y=y_{1,Cub_j^{fi}}}^{m=N-1,y=y_{2,Cub_j^{fi}}}$, $0 \le m \le N-1$, $y_{1,Cub_j^{fi}} \le y \le y_{2,Cub_j^{fi}}$

$I = \left\{f_i(x,y,z)\right\}_{x=x_{1,Cub_j^{fi}},y=y_{1,Cub_j^{fi}}}^{x=x_{2,Cub_j^{fi}},y=y_{2,Cub_j^{fi}}}$, $x_{1,Cub_j^{fi}} \le x \le x_{2,Cub_j^{fi}}$, $y_{1,Cub_j^{fi}} \le y \le y_{2,Cub_j^{fi}}$

- **Step 1 :** Computation K_j matrix along the z-direction by (IV.24), we can obtain temporary matrices of Meixner moments at each block of cuboid j, where $MM_{nm}^{b_{ij}^z}$ denotes the element of n^{th} row and m^{th} column in the k^{th} plane of z-direction.

$$MM_{nm}^{b_{ij}^z}(z), \quad z = z_{1,Cub_j^{fi}}, z_{1,Cub_j^{fi}}+1, \cdots, z_{2,Cub_j^{fi}}-1, z_{2,Cub_j^{fi}}$$

$$\begin{bmatrix} MM_{00}^{b_{ij}^{z2}}\left(z_{2,Cub_j^{fi}}\right) & MM_{01}^{b_{ij}^{z2}}\left(z_{2,Cub_j^{fi}}\right) & \cdots & MM_{0m}^{b_{ij}^{z2}}\left(z_{2,Cub_j^{fi}}\right) \\ MM_{10}^{b_{ij}^{z2}}\left(z_{2,Cub_j^{fi}}\right) & MM_{11}^{b_{ij}^{z2}}\left(z_{2,Cub_j^{fi}}\right) & \cdots & MM_{1m}^{b_{ij}^{z2}}\left(z_{2,Cub_j^{fi}}\right) \\ \vdots & \vdots & \ddots & \vdots \\ MM_{n0}^{b_{ij}^{z2}}\left(z_{2,Cub_j^{fi}}\right) & MM_{n1}^{b_{ij}^{z2}}\left(z_{2,Cub_j^{fi}}\right) & \cdots & MM_{nm}^{b_{ij}^{z2}}\left(z_{2,Cub_j^{fi}}\right) \end{bmatrix}$$

$$\vdots$$

$$\vdots$$

$$\vdots$$

$$\begin{bmatrix} MM_{00}^{b_{ij}^{z1+1}}\left(z_{1,Cub_j^{fi}}+1\right) & MM_{01}^{b_{ij}^{z1+1}}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & MM_{0m}^{b_{ij}^{z1+1}}\left(z_{1,Cub_j^{fi}}+1\right) \\ MM_{10}^{b_{ij}^{z1+1}}\left(z_{1,Cub_j^{fi}}+1\right) & MM_{11}^{b_{ij}^{z1+1}}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & MM_{1m}^{b_{ij}^{z1+1}}\left(z_{1,Cub_j^{fi}}+1\right) \\ \vdots & \vdots & \ddots & \vdots \\ MM_{n0}^{b_{ij}^{z1+1}}\left(z_{1,Cub_j^{fi}}+1\right) & MM_{n1}^{b_{ij}^{z1+1}}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & MM_{nm}^{b_{ij}^{z1+1}}\left(z_{1,Cub_j^{fi}}+1\right) \end{bmatrix}$$

$$\begin{bmatrix} MM_{00}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) & MM_{01}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) & \cdots & MM_{0m}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) \\ MM_{10}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) & MM_{11}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) & \cdots & MM_{1m}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) \\ \vdots & \vdots & \ddots & \vdots \\ MM_{n0}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) & MM_{n1}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) & \cdots & MM_{nm}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) \end{bmatrix}$$

- **Step 2:** In the second step, we rearrange the matrices obtained in Step 1 and multiplied by the polynomial matrices $P_3 = \left\{\tilde{M}_k^{(\gamma,\mu)}(z)\right\}_{k=0,z=z_{1,Cub_j^{fi}}}^{k=K-1,z=z_{2,Cub_j^{fi}}}$, $0 \le k \le K-1$, $z_{1,Cub_j^{fi}} \le z \le z_{2,Cub_j^{fi}}$ and get a $(n+1)$ matrix with size of $(m+1)\times(k+1)$, to obtain the 3D Meixner moments for each cuboid at the order (n,m,k).

Les moments Meixner 3D de chaque cuboïde peuvent être écrits avec la matrice suivante, pour l'ordre $n=0$:

$$\begin{bmatrix} MM_{000}^{Cub_j^{fi}} & MM_{001}^{Cub_j^{fi}} & \cdots & MM_{00k}^{Cub_j^{fi}} \\ MM_{010}^{Cub_j^{fi}} & MM_{011}^{Cub_j^{fi}} & \cdots & MM_{01k}^{Cub_j^{fi}} \\ \vdots & \vdots & \ddots & \vdots \\ MM_{0m0}^{Cub_j^{fi}} & MM_{0m1}^{Cub_j^{fi}} & \cdots & MM_{0mk}^{Cub_j^{fi}} \end{bmatrix} = \begin{bmatrix} \tilde{M}_0^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}\right) & \tilde{M}_0^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_0^{(\gamma,\mu)}\left(z_{2,Cub_j^{fi}}\right) \\ \tilde{M}_1^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}\right) & \tilde{M}_1^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_1^{(\gamma,\mu)}\left(z_{2,Cub_j^{fi}}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{M}_k^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}\right) & \tilde{M}_k^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_k^{(\gamma,\mu)}\left(z_{2,Cub_j^{fi}}\right) \end{bmatrix}$$
$$\times \begin{bmatrix} MM_{00}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}\right) & MM_{01}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}\right) & \cdots & MM_{0m}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}\right) \\ MM_{00}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}+1\right) & MM_{01}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & MM_{0m}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}+1\right) \\ \vdots & \vdots & \ddots & \vdots \\ MM_{00}^{b_{ij}^{\bar{z}1}}\left(z_{2,Cub_j^{fi}}\right) & MM_{01}^{b_{ij}^{\bar{z}1}}\left(z_{2,Cub_j^{fi}}\right) & \cdots & MM_{0m}^{b_{ij}^{\bar{z}1}}\left(z_{2,Cub_j^{fi}}\right) \end{bmatrix}$$
$$\vdots$$

Les moments Meixner 3D de chaque cuboïde peuvent être écrits avec la matrice suivante, pour l'order $n=1$:

$$\begin{bmatrix} MM_{100}^{Cub_j^{fi}} & MM_{101}^{Cub_j^{fi}} & \cdots & MM_{10k}^{Cub_j^{fi}} \\ MM_{110}^{Cub_j^{fi}} & MM_{111}^{Cub_j^{fi}} & \cdots & MM_{11k}^{Cub_j^{fi}} \\ \vdots & \vdots & \ddots & \vdots \\ MM_{1m0}^{Cub_j^{fi}} & MM_{1m1}^{Cub_j^{fi}} & \cdots & MM_{1mk}^{Cub_j^{fi}} \end{bmatrix} = \begin{bmatrix} \tilde{M}_0^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}\right) & \tilde{M}_0^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_0^{(\gamma,\mu)}\left(z_{2,Cub_j^{fi}}\right) \\ \tilde{M}_1^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}\right) & \tilde{M}_1^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_1^{(\gamma,\mu)}\left(z_{2,Cub_j^{fi}}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{M}_k^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}\right) & \tilde{M}_k^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_k^{(\gamma,\mu)}\left(z_{2,Cub_j^{fi}}\right) \end{bmatrix}$$
$$\times \begin{bmatrix} MM_{10}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}\right) & MM_{11}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}\right) & \cdots & MM_{1m}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}\right) \\ MM_{10}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}+1\right) & MM_{11}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & MM_{1m}^{b_{ij}^{\bar{z}1}}\left(z_{1,Cub_j^{fi}}+1\right) \\ \vdots & \vdots & \ddots & \vdots \\ MM_{10}^{b_{ij}^{\bar{z}1}}\left(z_{2,Cub_j^{fi}}\right) & MM_{11}^{b_{ij}^{\bar{z}1}}\left(z_{2,Cub_j^{fi}}\right) & \cdots & MM_{1m}^{b_{ij}^{\bar{z}1}}\left(z_{2,Cub_j^{fi}}\right) \end{bmatrix}$$
$$\vdots$$
$$\vdots$$
$$\vdots$$

Les moments Meixner 3D de chaque cuboïde peuvent être écrits avec la matrice suivante, pour l'order n :

$$\begin{bmatrix} MM_{n00}^{Cub_j^{fi}} & MM_{n01}^{Cub_j^{fi}} & \cdots & MM_{n0k}^{Cub_j^{fi}} \\ MM_{n10}^{Cub_j^{fi}} & MM_{n11}^{Cub_j^{fi}} & \cdots & MM_{n1k}^{Cub_j^{fi}} \\ \vdots & \vdots & \ddots & \vdots \\ MM_{nm0}^{Cub_j^{fi}} & MM_{nm1}^{Cub_j^{fi}} & \cdots & MM_{nmk}^{Cub_j^{fi}} \end{bmatrix} = \begin{bmatrix} \tilde{M}_0^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}\right) & \tilde{M}_0^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_0^{(\gamma,\mu)}\left(z_{2,Cub_j^{fi}}\right) \\ \tilde{M}_1^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}\right) & \tilde{M}_1^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_1^{(\gamma,\mu)}\left(z_{2,Cub_j^{fi}}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{M}_k^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}\right) & \tilde{M}_k^{(\gamma,\mu)}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_k^{(\gamma,\mu)}\left(z_{2,Cub_j^{fi}}\right) \end{bmatrix}$$
$$\times \begin{bmatrix} MM_{n0}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) & MM_{n1}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) & \cdots & MM_{nm}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}\right) \\ MM_{n0}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}+1\right) & MM_{n1}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}+1\right) & \cdots & MM_{nm}^{b_{ij}^{z1}}\left(z_{1,Cub_j^{fi}}+1\right) \\ \vdots & \vdots & \ddots & \vdots \\ MM_{n0}^{b_{ij}^{z1}}\left(z_{2,Cub_j^{fi}}\right) & MM_{n1}^{b_{ij}^{z1}}\left(z_{2,Cub_j^{fi}}\right) & \cdots & MM_{nm}^{b_{ij}^{z1}}\left(z_{2,Cub_j^{fi}}\right) \end{bmatrix}$$

La transformation inverse des moments de Meixner pour chaque cuboïde de la formule (IV.21) peut être réécrite comme suit:

$$f^{Cub_j^{fi}}(x,y,z) = \sum_{m=0}^{Max} \tilde{M}_m^{(\gamma,\mu)}(y) \times \left[\sum_{n=0}^{Max} \sum_{k=0}^{Max} MM_{nmk} \tilde{M}_n^{(\gamma,\mu)}(x) \tilde{M}_k^{(\gamma,\mu)}(z) \right] , \begin{cases} x_{1,Cub_j^{fi}} \le x \le x_{2,Cub_j^{fi}} \\ y_{1,Cub_j^{fi}} \le y \le y_{2,Cub_j^{fi}} \\ z_{1,Cub_j^{fi}} \le z \le z_{2,Cub_j^{fi}} \end{cases}$$
$$= \sum_{m=0}^{Max} \tilde{M}_m^{(\gamma,\mu)}(y) \times F^{b_{ij}^z}(x,z) \qquad \textbf{(IV.25)}$$

with $F^{b_{ij}^z}$ the inverse transform of Meixner moments in each block of a cuboid j, can also be written as:

$$F^{b_{ij}^z}(x,z) = \sum_{n=0}^{Max} \sum_{k=0}^{Max} MM_{nmk} \tilde{M}_k^{(\gamma,\mu)}(x) \tilde{M}_k^{(\gamma,\mu)}(z) \qquad \textbf{(IV.26)}$$

We calculate $F^{b_{ij}^z}$ in Eq. (26) by using a matrix form for a 2D image which can be re-expressed as follows:

$$F^{b_{ij}^z} = Q_1^T MM Q_2 \qquad \textbf{(IV.27)}$$

()T denotes the transpose of the matrix and $F^{b_{ij}^z}$, MM, Q_1^T and Q_2 are defined as:

$$F^{b_{ij}^z} = \left\{ F_{xz}^{b_{ij}^z} \right\}_{x=x_{1,Cub_j^{fi}},z=z_{1,Cub_j^{fi}}}^{x=x_{2,Cub_j^{fi}},z=z_{2,Cub_j^{fi}}} , \begin{cases} x_{1,Cub_j^{fi}} \le x \le x_{2,Cub_j^{fi}} \\ z_{1,Cub_j^{fi}} \le z \le z_{2,Cub_j^{fi}} \end{cases}$$

$$MM = \left\{ MM_{nmk} \right\}_{n,k=0}^{n,k=Max} , 0 \le n,k \le Max \quad , for\ each\ m\ with\ 0 \le m \le Max$$

$$Q_1 = \left\{ \tilde{M}_n^{(\gamma,\mu)}(x) \right\}_{n=0,x=x_{1,Cub_j^{fi}}}^{n=Max,x=x_{2,Cub_j^{fi}}} , \begin{cases} x_{1,Cub_j^{fi}} \le x \le x_{2,Cub_j^{fi}} \\ 0 \le n \le Max \end{cases}$$

$$Q_2 = \left\{ \tilde{M}_k^{(\gamma,\mu)}(z) \right\}_{k=0,z=z_{1,Cub_j^{fi}}}^{k=Max,z=z_{2,Cub_j^{fi}}} , \begin{cases} z_{1,Cub_j^{fi}} \le z \le z_{2,Cub_j^{fi}} \\ 0 \le k \le Max \end{cases}$$

- **Step 1:** Computation K_j matrix along the m-direction by (27), we can obtain temporary matrix the 2-D inverse transform of Meixner moments in each block of cuboid j, where $F_{xz}^{b_{ij}^z}$ denotes the element of xth row and zth column in the yth plane of m-direction.

$$F_{xz}^{b_{ij}^z}(m), \quad m = 0,1,\cdots, Max$$

$$\begin{bmatrix} F_{x_1,z_1}^{b_{ij}^z}(Max) & F_{x_1,z_1+1}^{b_{ij}^z}(Max) & \cdots & F_{x_1,z_2}^{b_{ij}^z}(Max) \\ F_{x_1+1,z_1}^{b_{ij}^z}(Max) & F_{x_1+1,z_2}^{b_{ij}^z}(Max) & \cdots & F_{x_1+1,z_2}^{b_{ij}^z}(Max) \\ \vdots & \vdots & \ddots & \vdots \\ F_{x_2,z_1}^{b_{ij}^z}(Max) & F_{x_2,z_1+1}^{b_{ij}^z}(Max) & \cdots & F_{x_2,z_2}^{b_{ij}^z}(Max) \end{bmatrix}$$

$$\vdots$$

$$\begin{bmatrix} F_{x_1,z_1}^{b_{ij}^z}(1) & F_{x_1,z_1+1}^{b_{ij}^z}(1) & \cdots & F_{x_1,z_2}^{b_{ij}^z}(1) \\ F_{x_1+1,z_1}^{b_{ij}^z}(1) & F_{x_1+1,z_2}^{b_{ij}^z}(1) & \cdots & F_{x_1+1,z_2}^{b_{ij}^z}(1) \\ \vdots & \vdots & \ddots & \vdots \\ F_{x_2,z_1}^{b_{ij}^z}(1) & F_{x_2,z_1+1}^{b_{ij}^z}(1) & \cdots & F_{x_2,z_2}^{b_{ij}^z}(1) \end{bmatrix}$$

$$\begin{bmatrix} F_{x_1,z_1}^{b_{ij}^z}(0) & F_{x_1,z_1+1}^{b_{ij}^z}(0) & \cdots & F_{x_1,z_2}^{b_{ij}^z}(0) \\ F_{x_1+1,z_1}^{b_{ij}^z}(0) & F_{x_1+1,z_2}^{b_{ij}^z}(0) & \cdots & F_{x_1+1,z_2}^{b_{ij}^z}(0) \\ \vdots & \vdots & \ddots & \vdots \\ F_{x_2,z_1}^{b_{ij}^z}(0) & F_{x_2,z_1+1}^{b_{ij}^z}(0) & \cdots & F_{x_2,z_2}^{b_{ij}^z}(0) \end{bmatrix}$$

- **Step 2 :** In the second step, we rearrange the matrices obtained in Step 1 and multiplied by the polynomial matrices $Q_3 = \left\{ \tilde{M}_m^{(\gamma,\mu)}(y) \right\}_{m=0,y=y_{1,Cub_j^{f_i}}}^{m=Max,y=y_{2,Cub_j^{f_i}}}$, $y_{1,Cub_j^{f_i}} \le y \le y_{2,Cub_j^{f_i}}$, $0 \le m \le Max$ and get a N_j matrix with size of $M_j \times K_j$, to obtain the inverse transform of Meixner Moments for each cuboid of size $\left(N_j, M_j, K_j\right)$.

The inverse transform of Meixner Moments for each cuboid can be written by the following matrices:

$$\begin{bmatrix} f^{Cub_j^{f_i}}(x_1,y_1,z_1) & f^{Cub_j^{f_i}}(x_1,y_1,z_1+1) & \cdots & f^{Cub_j^{f_i}}(x_1,y_1,z_2) \\ f^{Cub_j^{f_i}}(x_1,y_1+1,z_1) & f^{Cub_j^{f_i}}(x_1,y_1+1,z_1+1) & \cdots & f^{Cub_j^{f_i}}(x_1,y_1+1,z_2) \\ \vdots & \vdots & \ddots & \vdots \\ f^{Cub_j^{f_i}}(x_1,y_2,z_1) & f^{Cub_j^{f_i}}(x_1,y_2,z_1+1) & \cdots & f^{Cub_j^{f_i}}(x_1,y_2,z_2) \end{bmatrix} =$$

$$\begin{bmatrix} \tilde{M}_0^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}\right) & \tilde{M}_0^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}+1\right) & \cdots & \tilde{M}_0^{(\gamma,\mu)}\left(y_{2,Cub_j^{f_i}}\right) \\ \tilde{M}_1^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}\right) & \tilde{M}_1^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}+1\right) & \cdots & \tilde{M}_1^{(\gamma,\mu)}\left(y_{2,Cub_j^{f_i}}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{M}_{Max}^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}\right) & \tilde{M}_{Max}^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}+1\right) & \cdots & \tilde{M}_{Max}^{(\gamma,\mu)}\left(y_{2,Cub_j^{f_i}}\right) \end{bmatrix}^T \times \begin{bmatrix} F_{x_1,z_1}^{b_{ij}^z}(0) & F_{x_1,z_1+1}^{b_{ij}^z}(0) & \cdots & F_{x_1,z_2}^{b_{ij}^z}(Max) \\ F_{x_1,z_1}^{b_{ij}^z}(1) & F_{x_1,z_1+1}^{b_{ij}^z}(1) & \cdots & F_{x_1,z_2}^{b_{ij}^z}(Max) \\ \vdots & \vdots & \ddots & \vdots \\ F_{x_1,z_1}^{b_{ij}^z}(Max) & F_{x_1,z_1+1}^{b_{ij}^z}(Max) & \cdots & F_{x_1,z_2}^{b_{ij}^z}(Max) \end{bmatrix}$$

$$\vdots$$

$$\begin{bmatrix} f^{Cub_j^{f_i}}(x_1+1,y_1,z_1) & f^{Cub_j^{f_i}}(x_1+1,y_1,z_1+1) & \cdots & f^{Cub_j^{f_i}}(x_1+1,y_1,z_2) \\ f^{Cub_j^{f_i}}(x_1+1,y_1+1,z_1) & f^{Cub_j^{f_i}}(x_1+1,y_1+1,z_1+1) & \cdots & f^{Cub_j^{f_i}}(x_1+1,y_1+1,z_2) \\ \vdots & \vdots & \ddots & \vdots \\ f^{Cub_j^{f_i}}(x_1+1,y_2,z_1) & f^{Cub_j^{f_i}}(x_1+1,y_2,z_1+1) & \cdots & f^{Cub_j^{f_i}}(x_1+1,y_2,z_2) \end{bmatrix} =$$

$$\begin{bmatrix} \tilde{M}_0^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}\right) & \tilde{M}_0^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}+1\right) & \cdots & \tilde{M}_0^{(\gamma,\mu)}\left(y_{2,Cub_j^{f_i}}\right) \\ \tilde{M}_1^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}\right) & \tilde{M}_1^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}+1\right) & \cdots & \tilde{M}_1^{(\gamma,\mu)}\left(y_{2,Cub_j^{f_i}}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{M}_{Max}^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}\right) & \tilde{M}_{Max}^{(\gamma,\mu)}\left(y_{1,Cub_j^{f_i}}+1\right) & \cdots & \tilde{M}_{Max}^{(\gamma,\mu)}\left(y_{2,Cub_j^{f_i}}\right) \end{bmatrix} \times \begin{bmatrix} F_{x_1+1,z_1}^{b_{ij}^z}(0) & F_{x_1+1,z_1+1}^{b_{ij}^z}(0) & \cdots & F_{x_1,+1z_2}^{b_{ij}^z}(Max) \\ F_{x_1+1,z_1}^{b_{ij}^z}(1) & F_{x_1+1,z_1+1}^{b_{ij}^z}(1) & \cdots & F_{x_1+1,z_2}^{b_{ij}^z}(Max) \\ \vdots & \vdots & \ddots & \vdots \\ F_{x_1+1,z_1}^{b_{ij}^z}(Max) & F_{x_1+1,z_1+1}^{b_{ij}^z}(Max) & \cdots & F_{x_1+1,z_2}^{b_{ij}^z}(Max) \end{bmatrix}$$

$$\begin{bmatrix} f^{Cub_j^{fi}}(x_2,y_1,z_1) & f^{Cub_j^{fi}}(x_2,y_1,z_1+1) & \cdots & f^{Cub_j^{fi}}(x_2,y_1,z_2) \\ f^{Cub_j^{fi}}(x_2,y_1+1,z_1) & f^{Cub_j^{fi}}(x_2,y_1+1,z_1+1) & \cdots & f^{Cub_j^{fi}}(x_2,y_1+1,z_2) \\ \vdots & \vdots & \ddots & \vdots \\ f^{Cub_j^{fi}}(x_2,y_2,z_1) & f^{Cub_j^{fi}}(x_2,y_2,z_1+1) & \cdots & f^{Cub_j^{fi}}(x_2,y_2,z_2) \end{bmatrix} =$$

$$\begin{bmatrix} \tilde{M}_0^{(\gamma,\mu)}\left(y_{1,Cub_j^{fi}}\right) & \tilde{M}_0^{(\gamma,\mu)}\left(y_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_0^{(\gamma,\mu)}\left(y_{2,Cub_j^{fi}}\right) \\ \tilde{M}_1^{(\gamma,\mu)}\left(y_{1,Cub_j^{fi}}\right) & \tilde{M}_1^{(\gamma,\mu)}\left(y_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_1^{(\gamma,\mu)}\left(y_{2,Cub_j^{fi}}\right) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{M}_{Max}^{(\gamma,\mu)}\left(y_{1,Cub_j^{fi}}\right) & \tilde{M}_{Max}^{(\gamma,\mu)}\left(y_{1,Cub_j^{fi}}+1\right) & \cdots & \tilde{M}_{Max}^{(\gamma,\mu)}\left(y_{2,Cub_j^{fi}}\right) \end{bmatrix}^T \times \begin{bmatrix} F^{b_{ij}^{\xi}}_{x_2,z_1}(0) & F^{b_{ij}^{\xi}}_{x_2,z_1+1}(0) & \cdots & F^{b_{ij}^{\xi}}_{x_2,z_2}(Max) \\ F^{b_{ij}^{\xi}}_{x_2,z_1}(1) & F^{b_{ij}^{\xi}}_{x_2,z_1+1}(1) & \cdots & F^{b_{ij}^{\xi}}_{x_2,z_2}(Max) \\ \vdots & \vdots & \ddots & \vdots \\ F^{b_{ij}^{\xi}}_{x_2,z_1}(Max) & F^{b_{ij}^{\xi}}_{x_2,z_1+1}(Max) & \cdots & F^{b_{ij}^{\xi}}_{x_2,z_2}(Max) \end{bmatrix}$$

Enfin, nous fusionnons la transformation inverse des moments de Meixner de chaque cuboïde pour trouver l'image 3D reconstruite avec la même taille que l'image 3D d'origine.

Après avoir calculé les moments Meixner 3D de chaque cuboïde, nous avons additionné les matrices des moments de Meixner pour chaque cuboïde afin de trouver les moments de Meixner à partir de l'équation (IV.17). Le calcul direct et matriciel basé sur la représentation cuboïde d'image 3D (ICR) est plus rapide en termes de temps de calcul. Ainsi, la réduction significative en termes de temps de calcul des moments orthogonaux de Meixner par les deux approches proposées, permet d'appliquer ces méthodes de calcul sur plusieurs domaines de traitement d'images, tels que la reconstruction et la classification d'images. Dans la section suivante, nous discuterons la méthode proposée au moyen de simulations et prouverons l'efficacité de nos méthodes en termes de temps de calcul et de qualité de reconstruction.

5. Moments orthogonaux invariants à travers les cuboïdes

Dans cette section, nous définissons les moments orthogonaux invariants, nécessaires pour la classification des images, en vu d'apporter une contribution au niveau de la classification des images 3D à travers les moments orthogonaux de Meixner et la représentation des images 3D par cuboïdes.

Hu fut le premier à introduire les moments géométriques invariants pour la classification des images 2D en 1962 [Hu, 1962]. Il a utilisé la théorie des invariants algébriques pour définir sept moments invariants pour les transformations affines de la TSR

de translation, de changement d'échelle et de rotation en fonction des moments géométriques [H, 1962]. En conséquence, ces moments invariants sont largement utilisés comme attributs globaux pour la reconnaissance automatique de formes en raison de leur simplicité et de leur interprétation géométrique explicite.

Les moments sont appelés invariants, si leur expression est indépendante des paramètres de transformation TSR, de sorte que l'image doit être correctement reconnue, quelles que soient sa position et son orientation. En général, les moments invariants sont extraits par deux méthodes : la méthode directe et la méthode indirecte. Dans la méthode directe, également appelée méthode de normalisation d'image, on peut transformer l'image en une image standard ayant les propriétés de translation, changement d'échelle et de rotation par changements des variables (x, y) de l'image 2D et (x, y, z) de l'image 3D. Dans la méthode indirecte, nous calculons les moments orthogonaux invariants comme une combinaison linéaire de moments géométriques invariants. L'extraction de ces invariants est une tâche très difficile en termes de complexité et de temps de calcul.

Le calcul des moments invariants est un processus fastidieux et la complexité de calcul augmente en augmentant l'ordre des moments. Ainsi, des méthodes remarquables ont été introduites pour améliorer la précision et réduire les temps de calcul des moments invariants 2D et 3D de Legendre, Tchebichef, Krawtchouk, Hahn et Dual Hahn [Hosny, 2011 ; Upneja, 2015 ; Batioua, 2017; Sayyouri, 2014, 2015, 2016; Hmimid, 2014]. À notre connaissance, aucun travail n'a été consacré au calcul des moments invariants de Meixner pour les images 3D et à la classification des images 3D par les moments invariants de Meixner 3D.

Motivés par l'accélération du processus d'extraction des moments invariants de Meixner 3D et par l'amélioration des performances de classification d'objets 3D, nous proposons une nouvelle approche pour extraire les moments d'invariants de Meixner 3D et accélérer leur temps de calcul. Cette approche repose sur deux notions fondamentales : la première est l'extraction des moments invariants de Meixner 3D à partir des moments géométriques invariants 3D. La seconde est l'utilisation de la représentation d'images par cuboïde 3D (ICR). Cette représentation est proposée par les mêmes auteurs dans [Jahid, 2018 ; Karmouni, 2019] pour la reconstruction d'images 3D. Dans cette représentation, les

moments invariants de Meixner 3D seront extraits des cuboïdes au lieu de l'image globale, ce qui permet de réduire considérablement le temps de calcul des moments invariants de Meixner 3D et par conséquent de réduire le temps de classification des images 3D.

Ainsi, les résultats des simulations montrent que l'approche proposée est très efficace en termes de temps de calcul des moments invariants de Meixner 3D par rapport aux méthodes existantes. Ils montrent également la possibilité de classer les images 3D en fonction des moments invariants de Meixner en 3D par rapport à d'autres types de moments.

5.1. Moments géométriques invariants 3D

Dans cette section, nous introduisons des moments géométriques invariants 3D par rapport aux transformations spatiales dans le plan les plus simples, à la translation changement d'échelle et à la rotation (TSR). L'invariance du TSR est largement utilisée dans toutes les applications pratiques, car l'objet doit être correctement reconnu, quelles que soient sa position et son orientation.

L'ordre $(n+m+k)^{th}$ des moments géométriques 3D GM_{nmk} d'une fonction d'intensité d'image 3D $f(x,y,z)$, de taille $N\times M\times K$ est défini en utilisant une approximation de somme discrète comme [Flusser, 2016 ; Batioua, 2017]:

$$GM_{nmk}=\sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\sum_{z=0}^{K-1}x^n y^m z^k f(x,y,z) \tag{IV.28}$$

Ensuite, les moments centraux 3D μ_{nmk} sont des invariants de translation, peuvent être écrits comme suit [Batioua, 2017]

$$\mu_{nmk}=\sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\sum_{z=0}^{K-1}(x-\overline{x})^n (y-\overline{y})^m (z-\overline{z})^k f(x,y,z) \tag{IV.29}$$

Les coordonnées $(\overline{x},\overline{y},\overline{z})$ dénoter le centroïde de l'image $f(x,y,z)$ qui sont définis comme :

$$\bar{x} = \frac{GM_{100}}{GM_{000}}, \quad \bar{y} = \frac{GM_{010}}{GM_{000}} \quad \text{et} \quad \bar{z} = \frac{GM_{001}}{GM_{000}} \tag{IV.30}$$

La matrice de rotation peut être calculée à partir des conventions d'angle d'Euler par multiplication de matrice. Nous pouvons facilement déduire les matrices de rotation particulières autour d'axes individuels [Flusser, 2016].

La matrice de rotation autour de l'axe x par angle α est :

$$R_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix} \tag{IV.31}$$

la matrice de rotation autour de l'axe y par angle β est :

$$R_y(\beta) = \begin{pmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{pmatrix} \tag{IV.32}$$

et la matrice de rotation autour de l'axe z par angle γ est

$$R_z(\gamma) = \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{IV.33}$$

Les angles α, β, et γ sont les angles d'Euler.

où $\alpha = \frac{1}{2}\tan^{-1}\left(\frac{2\mu_{011}}{\mu_{020} - \mu_{002}}\right)$, $\beta = \frac{1}{2}\tan^{-1}\left(\frac{2\mu_{101}}{\mu_{200} - \mu_{002}}\right)$ **et** $\gamma = \frac{1}{2}\tan^{-1}\left(\frac{2\mu_{110}}{\mu_{200} - \mu_{020}}\right)$

Puisque la multiplication de matrice ne bascule pas, l'ordre des axes sur lesquels on tourne dépend du résultat. Pour cette analyse, nous allons faire d'abord une rotation autour de l'axe x, puis de l'axe y et enfin de l'axe z. Une telle séquence de rotations peut être représentée par le produit de la matrice [Batioua, 2017] :

$$R_{xyz}(\alpha, \beta, \gamma) = R_x(\alpha) R_y(\beta) R_z(\gamma) \tag{IV.34}$$

où

$$R_{xyz}(\alpha,\beta,\gamma)=\begin{pmatrix}1 & 0 & 0\\ 0 & \cos\alpha & -\sin\alpha\\ 0 & \sin\alpha & \cos\alpha\end{pmatrix}\begin{pmatrix}\cos\beta & 0 & -\sin\beta\\ 0 & 1 & 0\\ \sin\beta & 0 & \cos\beta\end{pmatrix}\begin{pmatrix}\cos\gamma & -\sin\gamma & 0\\ \sin\gamma & \cos\gamma & 0\\ 0 & 0 & 1\end{pmatrix} \quad \textbf{(IV.35)}$$

Tandis que

$$R_{xyz}(\alpha,\beta,\gamma)=\begin{pmatrix}\cos\beta\cos\gamma & -\cos\beta\sin\gamma & -\sin\beta\\ -\sin\alpha\sin\beta\cos\gamma+\cos\alpha\sin\gamma & \sin\alpha\sin\beta\sin\gamma+\cos\alpha\cos\gamma & -\sin\alpha\cos\beta\\ \cos\alpha\sin\beta\cos\gamma+\sin\alpha\sin\gamma & -\cos\alpha\sin\beta\sin\gamma+\sin\alpha\cos\gamma & \cos\alpha\cos\beta\end{pmatrix} \quad \textbf{(IV.36)}$$

Plus généralement, la translation et la rotation en 3D est une transformation linéaire, qui peut être décrite sous forme de matrice

$$\begin{pmatrix}x'\\ y'\\ z'\end{pmatrix}=R_{xyz}(\alpha,\beta,\gamma)\begin{pmatrix}x-\overline{x}\\ y-\overline{y}\\ z-\overline{z}\end{pmatrix}=\begin{pmatrix}a_{11} & a_{12} & a_{13}\\ a_{21} & a_{22} & a_{23}\\ a_{31} & a_{32} & a_{33}\end{pmatrix}\begin{pmatrix}x-\overline{x}\\ y-\overline{y}\\ z-\overline{z}\end{pmatrix} \quad \textbf{(IV.37)}$$

avec $\left(a_{ij}\right)_{\substack{1\le i\le m\\ 1\le j\le n}}$ sont les éléments de la matrice $R_{xyz}(\alpha,\beta,\gamma)$.

Ainsi, l'Eq. (10) peut être réécrit à l'aide de l'expression suivante :

$$\begin{pmatrix}x'\\ y'\\ z'\end{pmatrix}=\begin{pmatrix}a_{11}(x-\overline{x})+a_{12}(y-\overline{y})+a_{13}(z-\overline{z})\\ a_{21}(x-\overline{x})+a_{22}(y-\overline{y})+a_{23}(z-\overline{z})\\ a_{31}(x-\overline{x})+a_{32}(y-\overline{y})+a_{33}(z-\overline{z})\end{pmatrix} \quad \textbf{(IV.38)}$$

Ensuite, les moments géométriques invariants 3D d'ordre $(n+m+k)^{th}$ c'est noté GMI_{nmk}, qui sont indépendants de la translation, du changement d'échelle et de la rotation peuvent être écrits :

$$GMI_{nmk} = GM_{000}^{-\lambda} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \sum_{z=0}^{K-1} \begin{Bmatrix} \left(a_{11}\left(x-\bar{x}\right)+a_{12}\left(y-\bar{y}\right)+a_{13}\left(z-\bar{z}\right)\right)^{n} \\ \times\left(a_{21}\left(x-\bar{x}\right)+a_{22}\left(y-\bar{y}\right)+a_{23}\left(z-\bar{z}\right)\right)^{m} \\ \times\left(a_{31}\left(x-\bar{x}\right)+a_{32}\left(y-\bar{y}\right)+a_{33}\left(z-\bar{z}\right)\right)^{k} \end{Bmatrix} f\left(x,y,z\right) \quad \textbf{(IV.39)}$$

avec $\lambda = \dfrac{n+m+k}{3}+1$.

5.2. Moments Invariant de Meixner 3D

Dans cette section, nous dérivons les invariants de translation, changement d'échelle et de rotation (TSR) du calcul des moments de Meixner sous la forme d'une combinaison linéaire des moments géométriques invariants, à partir des polynômes de Meixner correspondants.

Les n^{th} polynômes de Meixner notés $M_n^{(\gamma,\mu)}(x)$ sont définis en termes de fonction hypergéométrique d'une variable x, à l'ordre n, défini dans l'intervalle $[0; N-1]$ comme indiqué dans [Papakostas, 2010d ; Nikiforov, 1991], avec γ et μ sont deux réels définis comme suit $\gamma \succ 0,\ 0 \prec \mu \prec 1$ respectent l'équation suivante :

$$M_n^{(\gamma,\mu)}(x) = (\gamma)_n \, {}_2F_1\left(-n,-x,\gamma;1-1/\mu\right) \quad \textbf{(IV.40)}$$

avec $x, n = 0,1,2,...,N-1$. ${}_2F_1$ est la fonction hypergéométrique généralisée donnée par :

$${}_2F_1\left(a,b;z\right) = \sum_{k=0}^{n} \frac{(a)_k (b)_k}{(c)_k} \frac{z^k}{k!} \quad \textbf{(IV.41)}$$

Le symbole $(a)_k$ est le symbole de Pochhammer défini comme

$$(a)_k = a(a+1)(a+2)...(a+k-1) = \frac{(a+k-1)!}{(a-1)!} \quad \textbf{(IV.42)}$$

Pour simplifier la notation, nous introduisons un autre symbole, la factorielle décroissant désignée par $\langle a \rangle_k$ défini comme

$$\langle a \rangle_k = (-1)^k (-a)_k = a(a-1)(a-2)...(a-k+1) = \frac{(a)!}{(a-k)!}$$
$$k \geq 1, \quad \langle a \rangle_0 = 1 \tag{IV.43}$$

$\langle x \rangle_k$ peut être étendu comme

$$\langle x \rangle_k = \sum_{i=0}^{k} S(k,i) x^i \tag{IV.44}$$

où $S(k,i)$ sont les nombres de Stirling du premier type, obtenus par la relation de récurrence suivante [Zhi, 2018]:

$$S(k,i) = S(k-1,i-1) - (k-1) S(k-1,i) \tag{IV.45}$$

avec $S(k,0) = S(0,i) = 0 \quad \forall k \geq 1 \ \ \forall i \geq 1$ and $S(0,0) = 1$

l'équation (IV.40) peut être exprimé en termes de nombres de Stirling $S(k,i)$ comme :

$$M_n^{(\gamma,\mu)}(x) = (\gamma)_n \sum_{k=0}^{n} \frac{(-n)_k}{(\gamma)_k \, k!} \left(\frac{1-\mu}{\mu} \right)^k \langle x \rangle_k$$
$$= (\gamma)_n \sum_{k=0}^{n} \sum_{i=0}^{k} \frac{(-n)_k}{(\gamma)_k \, k!} \left(\frac{1-\mu}{\mu} \right)^k S(k,i) x^i \tag{IV.46}$$

Ensuite, l'équation (IV.46) peut être réécrit comme :

$$M_n^{(\gamma,\mu)}(x) = \sum_{i=0}^{n} \alpha_{n,i}^{(\gamma,\mu)} x^i \tag{IV.47}$$

où

$$\alpha_{n,i}^{(\gamma,\mu)} = (\gamma)_n \sum_{k=i}^{n} \frac{(-n)_k}{(\gamma)_k\, k!} \left(\frac{1-\mu}{\mu}\right)^k S(k,i) \tag{IV.48}$$

L'ensemble des n polynômes de Meixner $\left\{M_n^{(\gamma,\mu)}(x)\right\}$ forme un ensemble complet de fonctions de base discrètes avec la fonction de pondération :

$$\omega(x) = \frac{\mu^x \Gamma(x+\gamma)}{\Gamma(x+1)\Gamma(\gamma)} \tag{IV.49}$$

et les polynômes de Meixner satisfont à la condition d'orthogonalité suivante :

$$\sum_{x=0}^{N-1} \omega(x) M_n^{(\gamma,\mu)}(x) M_m^{(\gamma,\mu)}(x) = \rho(n)\delta_{nm} \tag{IV.50}$$

où $n, m = 0,1,2,...,N-1$ et δ_{nm} est la fonction Kronecher.

tandis que $\rho(n)$ est la norme au carré exprimée par

$$\rho(n) = \frac{\Gamma(n+1)(\gamma)_n}{\mu^n (1-\mu)^\gamma} \tag{IV.51}$$

L'ensemble des polynômes de Meixner est orthogonal, mais il n'est pas orthonormé. Afin d'atteindre l'orthonormalité, les polynômes de Meixner normalisés par la norme $\rho(n)$ et le poids $\omega(x)$ peut dériver les polynômes de Meixner pondérés et normalisés $\left\{\tilde{M}_n^{(\gamma,\mu)}(x)\right\}$ défini par [Jahid, 2018a] :

$$\tilde{M}_n^{(\gamma,\mu)}(x) = M_n^{(\gamma,\mu)}(x) \sqrt{\frac{\omega(x)}{\rho(n)}} \tag{IV.52}$$

Par conséquent, la condition d'orthogonalité devient :

$$\sum_{x=0}^{N-1} \tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(x) = \delta_{nm} \tag{IV.53}$$

Les moments de Meixner 3D d'ordre $(n+m+k)^{th}$ en termes des polynômes orthogonaux discrets de Meixner, pour une fonction d'intensité d'image 3D $f(x,y,z)$ avec la taille $[0,\ N\text{-}1]\times[0,\ M\text{-}1]\times[0,\ K\text{-}1]$, sont définis sur le domaine [Jahid, 2018] :

$$MM_{nmk} = \sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\sum_{z=0}^{K-1} \tilde{M}_n^{(\gamma,\mu)}(x)\tilde{M}_m^{(\gamma,\mu)}(y)\tilde{M}_k^{(\gamma,\mu)}(z) f(x,y,z) \tag{IV.54}$$

avec $\tilde{M}_n^{(\gamma,\mu)}(x)$ est le n^{th} polynôme normalisé de Meixner.

Les moments Meixner 3D de l'image normalisée $\tilde{f}(x,y,z) = \left(\omega_k(x)\omega_k(y)\omega_k(z)\right)^{-1/2} f(x,y,z)$ de taille $(n+m+k)^{th}$ peut être écrit en termes des moments géométriques, en utilisant (IV.28) et (IV.47), comme suit :

$$\begin{aligned} MM_{nmk} &= \left[\rho(n)\rho(m)\rho(k)\right]^{-1/2} \sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\sum_{z=0}^{K-1} M_n^{(\gamma,\mu)}(x) M_m^{(\gamma,\mu)}(y) M_k^{(\gamma,\mu)}(z) f(x,y,z) \\ &= \left[\rho(n)\rho(m)\rho(k)\right]^{-1/2} \sum_{i=0}^{n}\sum_{j=0}^{m}\sum_{l=0}^{k} \alpha_{n,i}^{(\gamma,\mu)}\alpha_{m,j}^{(\gamma,\mu)}\alpha_{k,r}^{(\gamma,\mu)} GM_{ijr} \end{aligned} \tag{IV.55}$$

Les moments invariants de Meixner 3D sont obtenus en termes des moments invariants géométriques 3D. Par conséquent, nous pouvons obtenir les invariants de translation, de changement d'échelle et de rotation des moments de Meixner 3D d'ordre $(n+m+k)^{th}$, en remplaçant GM_{ijr} (moments géométriques) dans l'équation (IV.55) par GMI_{ijr} (moments invariants géométriques) définis dans l'équation (IV.39) :

$$MMI_{nmk} = \left[\rho(n)\rho(m)\rho(k)\right]^{-1/2} \sum_{i=0}^{n}\sum_{j=0}^{m}\sum_{l=0}^{k} \alpha_{n,i}^{(\gamma,\mu)}\alpha_{m,j}^{(\gamma,\mu)}\alpha_{k,r}^{(\gamma,\mu)} GMI_{ijr} \tag{IV.56}$$

où $\alpha_{n,i}^{(\gamma,\mu)}$ est le coefficient des monômes des polynômes orthogonaux discrets de Meixner définis par l'équation (IV.48).

L'algorithme suivant résume les étapes de calcul des moments invariants de Meixner 3D à partir des moments invariants géométriques 3D :

Algorithme: Calcul de MMI_{nmk} par la méthode classique
Entrée: Image 3D f ; des moments d'ordre $n,m,k \geq 0$; les paramètres de Meixner $\gamma \succ 0,\ 0 \prec \mu \prec 1$;
Étape 1: Calculer les GMI_{nmk} d'ordre $\left(n+m+k\right)^{th}$ en utilisant(IV.39) ;
Étape 2: Calculer les MMI_{nmk} d'ordre $\left(n+m+k\right)^{th}$ en utilisant **(IV.39) et (IV.55)**;
Sortie: Les MMI_{nmk} d'ordre $\left(n+m+k\right)^{th}$

Ainsi, l'application de l'algorithme présenté pour la classification des images 3D est limitée par le temps de calcul très élevé. Ainsi, pour accélérer le processus de sélection des primitives pour les moments invariants de Meixner 3D afin d'accélérer le processus de sélection des primitives pour les moments invariants Meixner 3D, nous proposons dans cette section une autre méthode pour dériver les moments invariants de Meixner, basée sur l'utilisation de Représentation d'images par cuboïdes 3D (ICR). Cette représentation a été introduite par les mêmes auteurs de cet article [38-39], afin d'accélérer le temps de calcul des moments de Charlier et Meixner et le temps de reconstruction des images 3D.

5.3. Calcul rapide des moments invariants de Meixner 3D

Dans cette section, nous présenterons la même représentation (ICR) pour l'accélération des moments invariants de Meixner et accélèrerons donc le temps de classification des images 3D.

Dans cette méthode, l'image 3D est décomposée en un ensemble de sous-images 3D d'intensité unique et chaque intensité est représentée par un ensemble de cuboïdes et chaque cuboïde correspondant à un objet. Ces cuboïdes sont définis comme un ensemble de voxels connectés. Cette méthode a l'avantage d'accélérer le processus d'extraction des moments

invariants de Meixner 3D et nous permet d'améliorer le résultat de la classification sans affecter la propriété d'invariance par rapport à la méthode classique.

En appliquant l'algorithme ICR [Karmouni, 2019], l'image 3D $f(x,y,z)$ contenant L cuboïdes, peut être définie comme

$$f(x,y,z)=\sum_{i=1}^{n_f} f_i(x,y,z) \tag{IV.57}$$

Avec $\{f_i,\ i=1,2,...,n_f\}$ est i^{th} d'intensité i de l'image 3D, et n_f est le nombre des valeurs d'intensité.

En utilisant l'algorithme ICR, les moments géométriques invariants 3D proposée d'ordre $(n+m+k)^{th}$ d'une image 3D $f(x,y,z)$ de taille $N\times M\times K$ dans l'équation (39), peut être réécrit comme

$$PGMI_{nmk}=GM_{000}^{-\lambda}\sum_{x=0}^{N-1}\sum_{y=0}^{M-1}\sum_{z=0}^{K-1}\left\{\begin{array}{l}\left(a_{11}(x-\overline{x})+a_{12}(y-\overline{y})+a_{13}(z-\overline{z})\right)^n\\ \times\left(a_{21}(x-\overline{x})+a_{22}(y-\overline{y})+a_{23}(z-\overline{z})\right)^m\\ \times\left(a_{31}(x-\overline{x})+a_{32}(y-\overline{y})+a_{33}(z-\overline{z})\right)^k\end{array}\right\}\sum_{i=1}^{n_f} f_i \tag{IV.58}$$

Les moments géométriques invariants 3D proposée dans (IV.58), peut être calculé comme suit

$$PGMI_{nmk}=\sum_{i=1}^{n_f} PGMI_{nmk}^{i} \tag{IV.59}$$

où $PGMI_{nmk}^{i}$ sont les moments géométriques invariants proposés calculés en considérant une intensité f_i d'image 3D, sont donnés par

$$PGMI^{i}_{nmk} = f_i \left[GM^{-\lambda}_{000} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \sum_{z=0}^{K-1} \begin{Bmatrix} \left(a_{11}\left(x-\overline{x}\right)+a_{12}\left(y-\overline{y}\right)+a_{13}\left(z-\overline{z}\right)\right)^{n} \\ \times\left(a_{21}\left(x-\overline{x}\right)+a_{22}\left(y-\overline{y}\right)+a_{23}\left(z-\overline{z}\right)\right)^{m} \\ \times\left(a_{31}\left(x-\overline{x}\right)+a_{32}\left(y-\overline{y}\right)+a_{33}\left(z-\overline{z}\right)\right)^{k} \end{Bmatrix} \right] \quad \textbf{(IV.60)}$$

Les moments géométriques invariants $PGMI^{i}_{nmk}$ correspondant à une intensité f_i sont calculés à partir des cuboïdes de chaque intensité. Puis l'ordre $\left(n+m+k\right)^{th}$ des L_i cuboïde, ayant des coordonnées $\left(x_{1,Cub^{f_i}_j}, x_{2,Cub^{f_i}_j}\right)$, $\left(y_{1,Cub^{f_i}_j}, y_{2,Cub^{f_i}_j}\right)$ et $\left(z_{1,Cub^{f_i}_j}, z_{2,Cub^{f_i}_j}\right)$, peut être calculé en utilisant la formule suivante :

$$\begin{aligned} PGMI^{i}_{nmk} &= \sum_{j=1}^{L_i} f_i \left[GM^{-\lambda}_{000} \sum_{x=x_{1,Cub^{f_i}_j}}^{x_{2,Cub^{f_i}_j}} \sum_{y=y_{1,Cub^{f_i}_j}}^{y_{2,Cub^{f_i}_j}} \sum_{z=z_{1,Cub^{f_i}_j}}^{z_{2,Cub^{f_i}_j}} \begin{Bmatrix} \left(a_{11}\left(x-\overline{x}\right)+a_{12}\left(y-\overline{y}\right)+a_{13}\left(z-\overline{z}\right)\right)^{n} \\ \times\left(a_{21}\left(x-\overline{x}\right)+a_{22}\left(y-\overline{y}\right)+a_{23}\left(z-\overline{z}\right)\right)^{m} \\ \times\left(a_{31}\left(x-\overline{x}\right)+a_{32}\left(y-\overline{y}\right)+a_{33}\left(z-\overline{z}\right)\right)^{k} \end{Bmatrix} \right] \\ &= \sum_{j=1}^{L_i} PGMI^{Cub^{f_i}_j}_{nmk} \end{aligned} \quad \textbf{(IV.61)}$$

où $Cub^{f_i}_j$ est le cuboïde j d'intensité f_i, $j=1,\cdots,L_i$ et $L=\sum_{i=1}^{n_f} L_i$, L_i le nombre de cuboïde pour chaque intensité f_i.

Tandis que $PGMI^{Cub^{f_i}_j}_{nmk}$ sont les moments géométriques invariants pour chaque cuboïde $Cub^{f_i}_j$, le calcul d'ordre $\left(n+m+k\right)^{th}$, ils sont définis comme

$$PGMI^{Cub^{f_i}_j}_{nmk} = GM^{-\lambda}_{000} \sum_{x=x_{1,Cub^{f_i}_j}}^{x_{2,Cub^{f_i}_j}} \sum_{y=y_{1,Cub^{f_i}_j}}^{y_{2,Cub^{f_i}_j}} \sum_{z=z_{1,Cub^{f_i}_j}}^{z_{2,Cub^{f_i}_j}} f_i \times \begin{Bmatrix} \left(a_{11}\left(x-\overline{x}\right)+a_{12}\left(y-\overline{y}\right)+a_{13}\left(z-\overline{z}\right)\right)^{n} \\ \times\left(a_{21}\left(x-\overline{x}\right)+a_{22}\left(y-\overline{y}\right)+a_{23}\left(z-\overline{z}\right)\right)^{m} \\ \times\left(a_{31}\left(x-\overline{x}\right)+a_{32}\left(y-\overline{y}\right)+a_{33}\left(z-\overline{z}\right)\right)^{k} \end{Bmatrix} \quad \textbf{(IV.62)}$$

Par conséquent, les moments géométriques invariants 3D proposés en translation, le changement d'échelle et la rotation d'image 3D peuvent être obtenus à partir de la somme des moments géométriques invariants 3D de chaque cuboïde $PGMI_{nmk}^{Cub_j^{f_i}}$ de chaque intensité f_i d'ordre $\left(n+m+k\right)^{th}$, peut être calculé comme

$$PGMI_{nmk} = \sum_{i=1}^{n_f} \sum_{j=1}^{L_i} PGMI_{nmk}^{Cub_j^{f_i}} \tag{IV.63}$$

Enfin, nous pouvons obtenir la translation, le changement d'échelle et la rotation des moments invariants de Meixner 3D d'ordre $\left(n+m+k\right)^{th}$, à partir les équations (IV.56) et (IV.63).

L'algorithme suivant résume les étapes de calcul des moments invariants de Meixner 3D à partir des moments géométriques invariants 3D proposés à l'aide de l'algorithme ICR.

Algorithme: Calcul rapide de $PMMI_{nmk}$ par algorithme ICR

Entrée: Image 3D f; Moments d'ordre $n,m,k \geq 0$; les paramètres de Meixner $\gamma \succ 0,\ 0 \prec \mu \prec 1$;

Étape 1 : Divise l'image 3D $f\left(x,y,z\right)$ à L cuboïdes par algorithme ICR [Jahid, 2018];

Étape 2 : Générer les $PGMI_{nmk}^{Cub_j^{f_i}}$ pour chaque cuboids $Cub_j^{f_i}$ de chaque intensité f_i en utilisant **(IV.62)**;

Étape 3 : Calculer les $PGMI_{nmk}$ d'ordre $\left(n+m+k\right)^{th}$ en utilisant **(IV.62)**;

Étape 4 : Calculer les $PMMI_{nmk}$ d'ordre $\left(n+m+k\right)^{th}$ en utilisant **(IV.56) et (IV.62)**;

Sortie : Les $PMMI_{nmk}$ d'ordre $\left(n+m+k\right)^{th}$

6. Simulations et résultats : Moments et Reconstruction d'images 3D

Afin d'évaluer les méthodes proposées, nous avons présenté dans cette section les simulations nécessaires pour appuyer l'intérêt porté à notre proposition. Dans cette section,

nous calculons les moments de Meixner et leurs inverses pour les images 3D choisis pour les tests, à travers plusieurs comparaisons, nous étudions l'efficacité de notre méthode sur le calcul des moments et de leurs inverses. Nous présentons d'abord des exemples d'images sur lesquelles nous avons appliqué l'algorithme proposé pour extraire les cuboïdes (ICR), puis nous exposons les temps de calcul nécessaires pour calculer les moments de Meixner et leurs inverses pour différents ordres. Nous établissons ensuite une comparaison entre plusieurs bases orthogonales de Tchebichef, Krawtchouk, Hahn et Charlier. Dans la deuxième partie de cette section, nous nous intéressons à l'utilisation de notre méthode pour la reconstruction d'image 3D à partir des moments de Meixner calculés. Nous comparons donc notre méthode de reconstruction à celle de la méthode récursive par rapport à la variable x largement utilisée pour la reconstruction en ce qui concerne les moments orthogonaux. Ensuite, nous étudions les résultats de la reconstruction d'images 3D en utilisant les polynômes de Tchebichef, Krawtchouk, Hahn, Charlier et Meixner à travers notre approche.

6.1. Extraction des cuboïdes sur une image 3D discrète

Ci-dessous, nous présentons les résultats obtenus, en termes de nombre de cuboïdes extraits de chaque image 3D, nous avons utilisé comme image de test cinq images 3D de taille $128\times128\times128$ voxels (Figure IV.4), choisies dans l'image de base "McGill" [Mcgill, 2017]. En tant que matrice tridimensionnelle, chaque image est composée d'un nombre différent de cuboïdes. Après l'extraction des cuboïdes de chaque image par l'application de l'algorithme ICR, le nombre de cuboïdes de chaque image est donné dans le tableau (IV.3), cette nouvelle représentation de l'image 3D réduit considérablement le nombre d'éléments à traiter.

Tableau (IV.3): nombre de cuboïdes extraits par image.

3D Images	Cup	Chair	Human	Octopuse	Spectacles
Number of cuboids	5066	205	823	1295	986

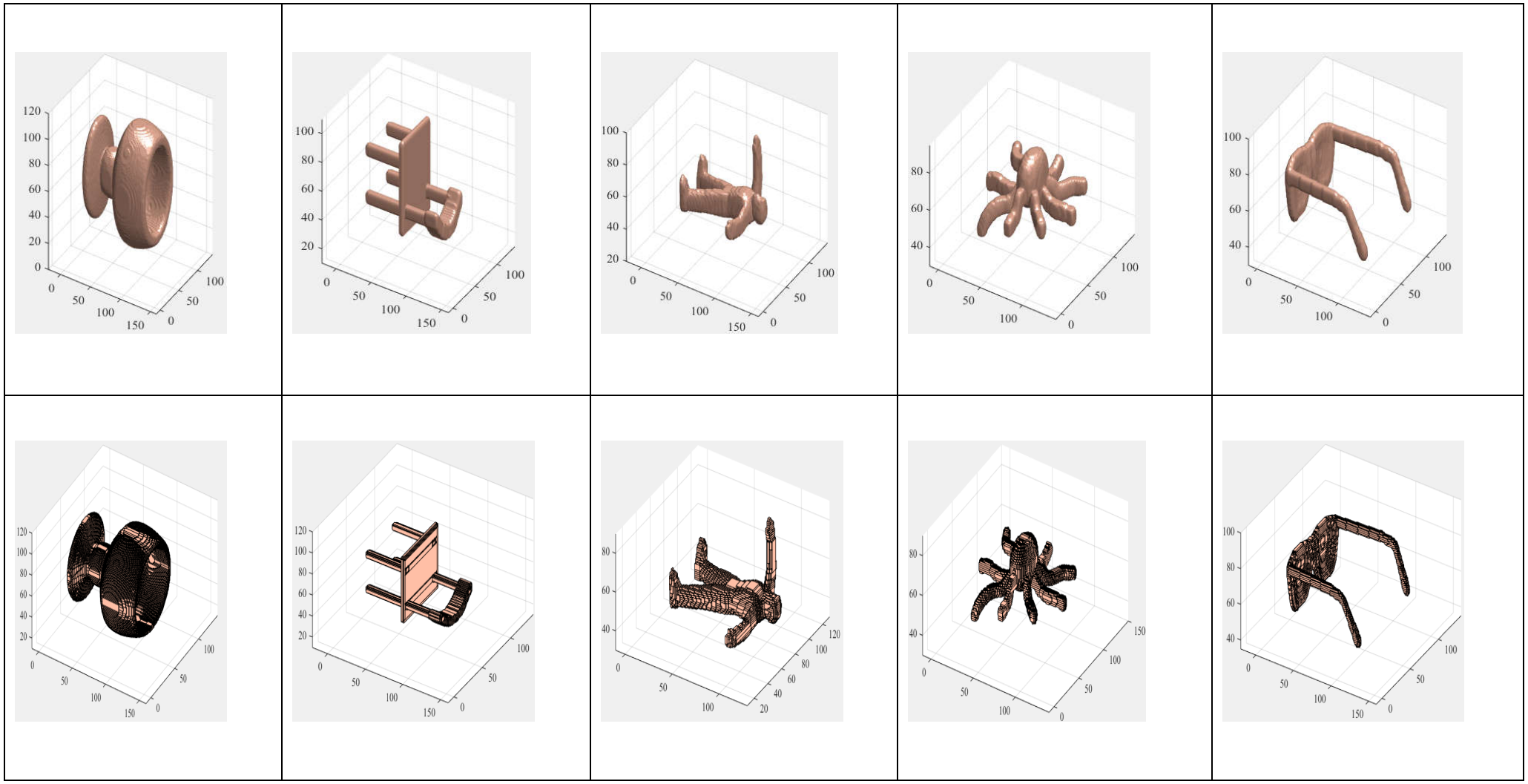
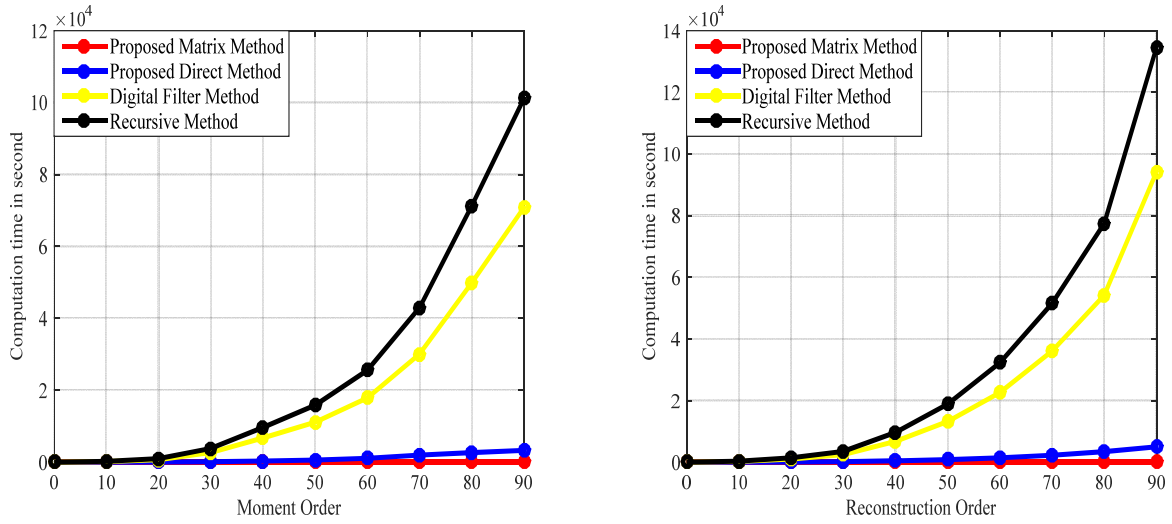


Figure (IV.4): Ensmble d'image 3D: Cup, Chair, Human, Octopuse and Spectacles de la base McGILL.

6.2. Temps de calcul des moments de Meixner et leurs inverses pour une image 3D

Le processus de calcul des moments orthogonaux discrets de Meixner 3D et de sa transformée inverse, est effectué pour des ordres allant de 0 à 90 pour cinq images 3D de dimensions $128\times128\times128$ illustrée sur le figure (IV.4). La figure (IV.5.a et b) représente respectivement le temps moyen de calcul des moments 3D et de leurs inverses par Meixner en utilisant la méthode récursive par rapport à *x*, la méthode du filtre numérique [Chang, 2012; Jahid, 2017; Karmouni, 2018] et la méthode proposée en mode direct (ICR) et indirect (ICR + par produit matriciel). Cette figure montre que la méthode proposée en version matricielle est plus rapide que la méthode directe proposée, la méthode du filtre numérique et la méthode récursive. Notez que le temps d'extraction du cuboïde de chaque image 3D ne dépasse pas 0,01 ms. Ce temps est bien inférieur au temps de calcul requis pour le calcul des moments orthogonaux 3D discrets à partir de l'image 3D.

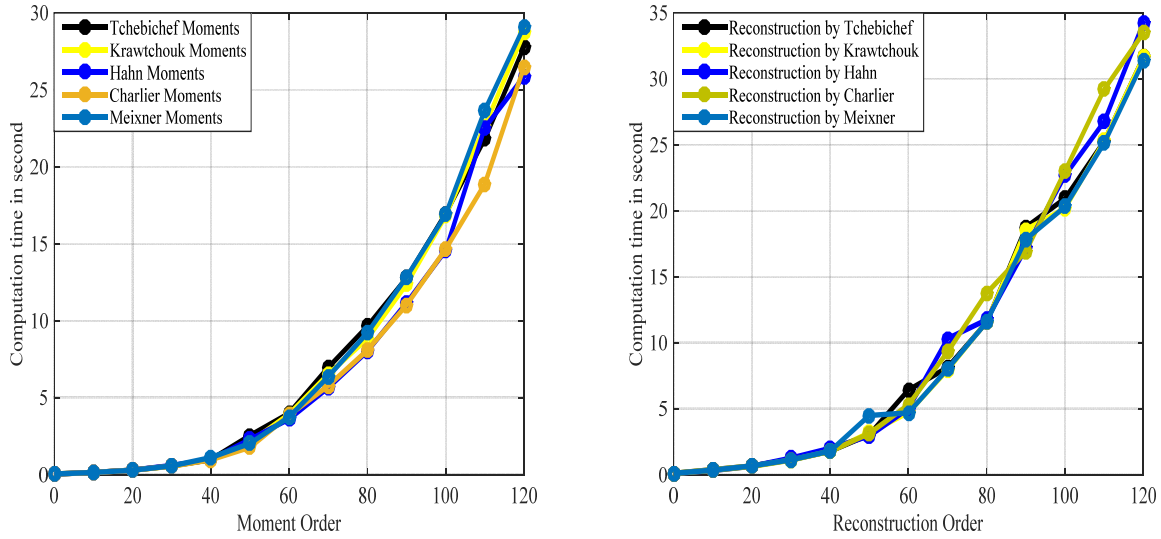


(a) (b)

Figure (IV.5): (a) Temps moyen pour le calcul des moments de Meixner pour des images 3D (b) Temps moyens pour calculer leurs inverses. en utilisant la méthode récursive et la méthode proposé Directe et par produit matriciel

6.3. Temps de calcul des moments orthogonaux discrets et leurs inverse pour d'autres bases

Pour comprendre l'apport de la méthode proposée pour calculer les moments orthogonaux et leurs inverses, nous avons appliqué cette dernière, sur les polynômes de Tchebichef, Krawtchouk, Hahn et de Charlier également, les deux figures (IV.6.a et b) ci-dessous représentent respectivement les temps consommés calculer les moments et leurs inverses à l'aide des deux polynômes en comparaison avec les polynômes de Meixner.



(a) (b)

Figure (IV.6): (a) Temps moyen de calcul des moments de Tchebichef, Krawtchouk, Hahn, Charlier and Meixner pour une image 3D. (b) Temps moyens de calcul de leurs inverses respectivement, en utilisant la méthode des cuboïdes

Nous pouvons seulement noter que les temps de calcul nécessaires pour calculer les moments et leurs inverses basés sur les différents polynômes orthogonaux via la méthode proposée sont si proches et évoluent identiquement, d'où la fiabilité et l'applicabilité de notre approche.

6.4. Reconstruction d'image 3D par les moments de Meixner

Afin de prouver l'efficacité et la robustesse de la méthode proposée pour la reconstruction d'images en 3D, comparons la méthode de reconstruction globale à la méthode de reconstruction locale basée sur les cuboïdes.

Dans ce qui suit, nous définissons des critères objectifs communément utilisés dans la littérature pour mesurer la qualité des images 3D reconstruites. Pour évaluer les performances de deux méthodes, nous allons calculer l'erreur quadratique moyenne (MSE) entre l'image d'origine et l'image reconstruite [Jahid, 2017 ; Karmouni, 2018]. Ce critère est largement utilisé dans le domaine de l'analyse d'images en tant que mesure quantitative de la précision de la reconstruction.

L'erreur quadratique moyenne pour une image 3D de taille $N \times M \times K$ est défini par:

$$MSE = \frac{1}{N \times M \times K} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \sum_{z=0}^{K-1} \left(f(x,y,z) - \hat{f}(x,y,z) \right)^2 \quad \textbf{(IV.64)}$$

Avec $\hat{f}(x,y,z)$ l'image 3D reconstruite pour chaque $f(x,y,z)$ de chaque voxel (x,y,z).

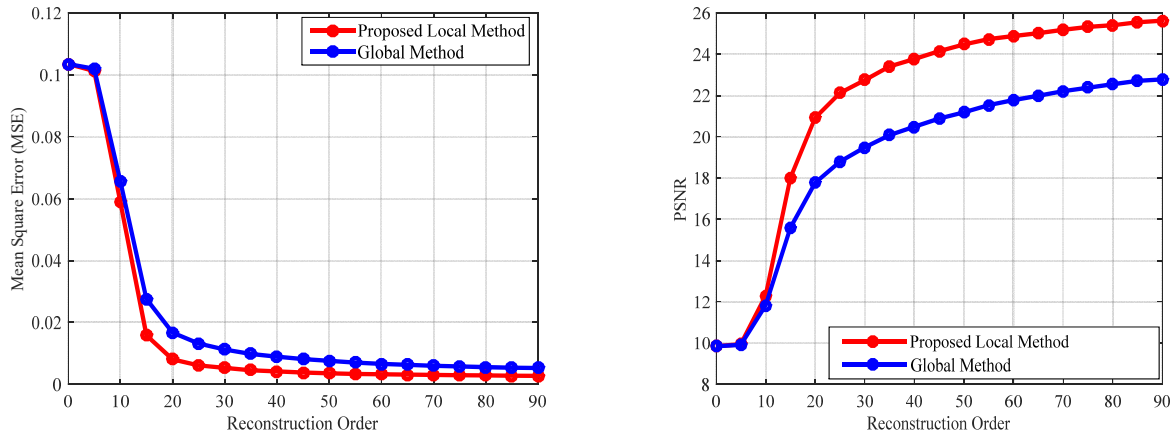
Dans [42], le Peak Signal-To-Noise Ratio est défini en decibels (d_B) comme suit:

$$PSNR = 10 \log_{10} \left(\frac{k^2}{MSE} \right) \quad \textbf{(IV.65)}$$

k : est la valeur maximale dans chaque voxel de l'image d'origine 3D.

Pour illustrer les performances de la méthode proposée, notre algorithme est testé sur une image 3D de taille $128 \times 128 \times 128$, représentant "Cup" (Figure IV.4) extraite de la base de données d'images 3D "McGill". Le test de reconstruction est effectué selon les deux méthodes: la méthode locale proposée et la méthode globale dans [Batioua, 2017] en utilisant des moments orthogonaux discrets 3D Meixner pour des ordres allant de 0 à 90.

La figure (IV.7) illustre l'erreur quadratique moyenne (MSE) et le rapport (signal / bruit) de pointe pour l'image 3D "Cup" par les deux méthodes. Nous notons que l'erreur quadratique moyenne MSE décroît à mesure que les ordres de Meixner 3D augmentent, où MSE s'approche de zéro avec l'augmentation de l'ordre des moments. Lorsque l'ordre maximal des moments atteint une certaine valeur, l'image reconstruite sera très proche de l'image d'origine. De même pour le rapport PNSR, qui augmente avec l'augmentation des ordres de moments Meixner jusqu'à obtenir une valeur stable où l'image reconstruite sera au maximum proche de l'image originale.



(a) (b)

Figure (IV.7): Mean squared error « MSE » et Peak Signal-To-Noise Ratio « PNSR » pour l'image reconstruite "Cup" Par la méthode proposé directe et avec produit matriciel

Comme le montrent les figures (IV.7.a et b), en utilisant respectivement MSE et PSNR, notre méthode proposée basée sur la représentation ICR améliore nettement la qualité de la reconstruction d'une image 3D discrète. Cela nous amène à avoir confiance en la robustesse de notre méthode, non seulement en termes de temps de calcul, mais également en qualité de reconstruction.

La figure (IV.8) montre l'image originale 3D "Cup" de taille $128\times128\times128$ et les images reconstruites par les moments de Meixner via deux méthodes utilisées "la méthode globale" et "la méthode locale proposée", à partir des ordres allant jusqu'à 20, 30, 40, 50. et 80.

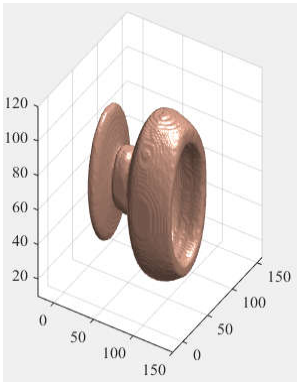
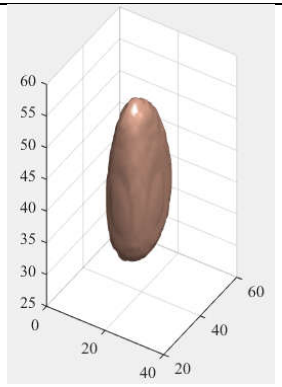
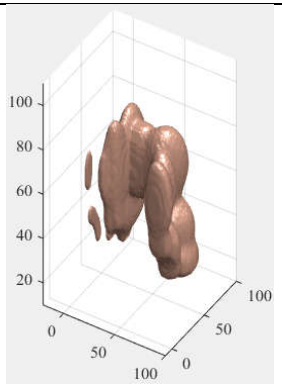
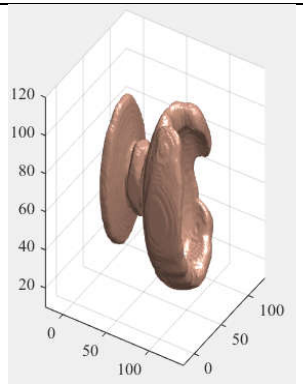
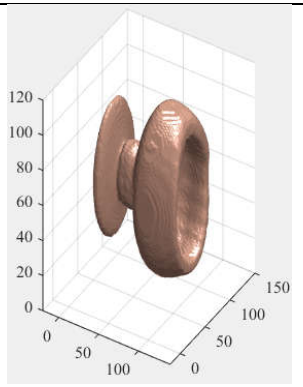
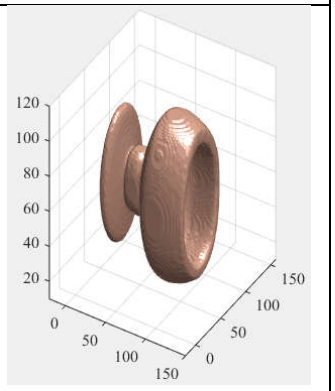
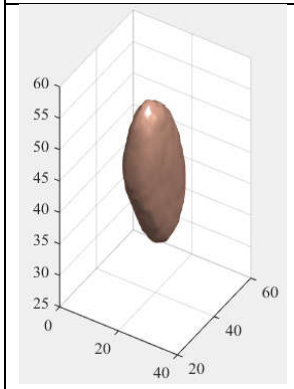
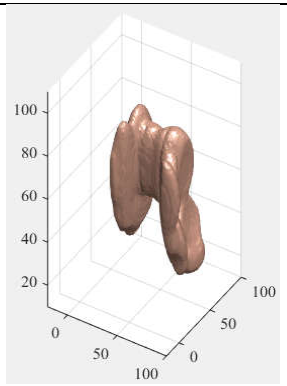
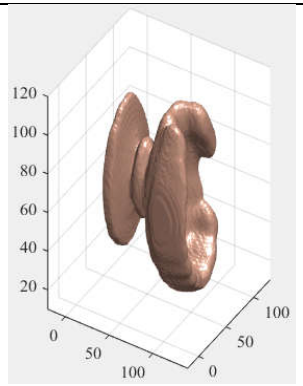
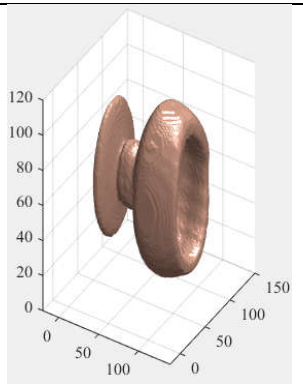
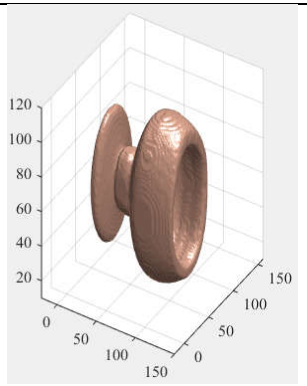
Order	Original image	20	30	40	50	80
Recursive Method						
Proposed Method						

Figure (IV.8): l'image reconstruite 3D " Cup " pour les ordres 20, 30, 40, 50, 80, respectivement par la récursive et la méthode proposé

L'index de mesure des structures de similitudes (Structural Similarity Index Measure) (SSIM) est utilisé fréquemment au lieu du MSE et du PSNR dans de nombreuses comparaisons perceptuelles [Wang, 2014]. En pratique, une seule mesure de la qualité globale de l'ensemble de l'image est généralement nécessaire pour évaluer la qualité globale d'une image reconstruite [Wang, 2009]. Cet index de mesure valide la fiabilité de notre approche, tel qu'on peut observer sur la figure (IV.9), l'image reconstruite s'approche de l'image originale pour des ordres plus faibles par rapport à ceux nécessaire pour la méthode classique.

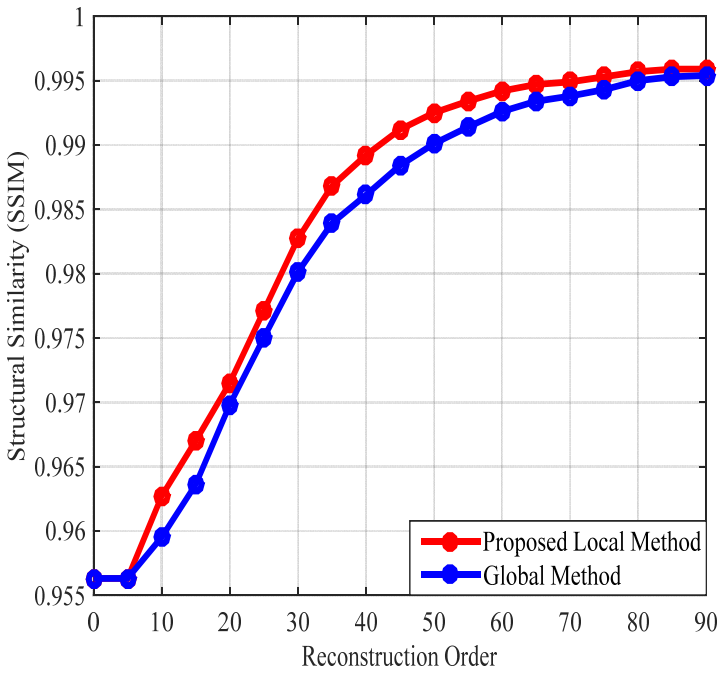


Figure (IV.9) : Structural Similarity « SSIM » pour image 3D "Cup" Par la méthode des cuboïdes

6.5. Reconstruction d'image 3D par les moments orthogonaux discrets : méthode proposé

Pour tester notre méthode, nous effectuons une reconstruction discrète d'images 3D à l'aide des moments orthogonaux de trois bases orthogonales Tchebichef, Krawtchouk, Hahn, Charlier et Meixner, en appliquant notre méthode proposée. Les simulations confirment que

les polynômes de Tchebichef, de Krawtchouk, et de Charlier donnent de meilleurs résultats, comme le démontrent le MSE et le PSNR de la reconstruction de l'image discrète 3D illustrée à la figure (IV.10.a et b).

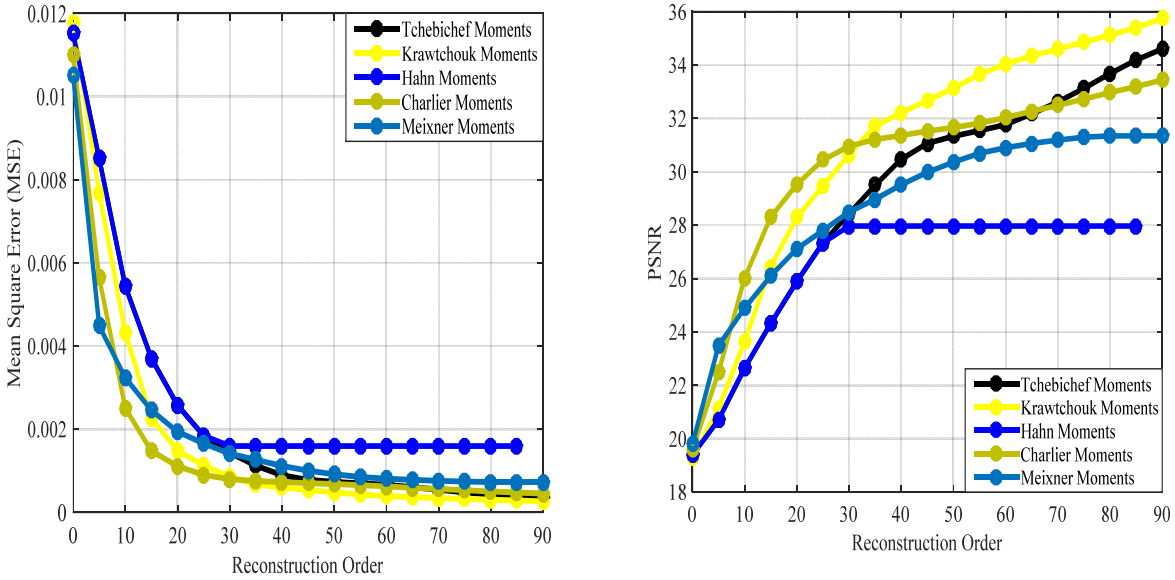


Figure (IV.10) (a) Mean squared error « MSE » and (b) Peak Signal-To-Noise Ratio « PNSR » pour l'image reconstruite « Cup » par les moments de Tchebichef, Krawtchouk, Hahn, Charlier and Meixner

Cette comparaison confirme que notre méthode est compatible avec différents polynômes orthogonaux. Nous pouvons dire que les polynômes de Meixner ne sont pas le meilleur choix pour une application de reconstruction d'image 3D, mais ceci n'est pas le cœur de notre proposition, nous pouvons en conclure que notre méthode permet effectivement d'améliorer considérablement le temps de calcul des moments ainsi que de réduire le taux d'erreur lors de la reconstruction de l'image 3D, nous pouvons également en conclure que les polynômes de Krawtchouk, principalement puis ceux de Tchebichef, sont plus approprié pour la reconstruction d'image 3D que celui de Meixner. En fait, nous devons explorer l'apport de ces moments dans la classification des images 3D pour en savoir plus sur les performances de chacun de ces polynômes, ce que nous attendons de nos futurs travaux.

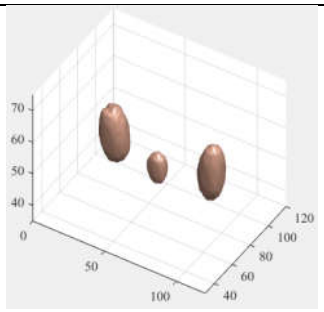
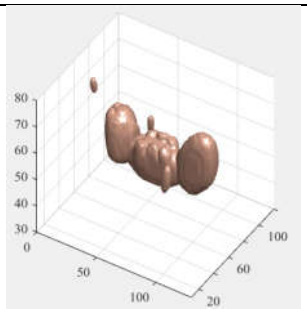
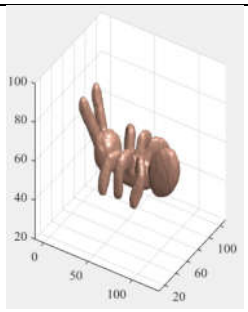
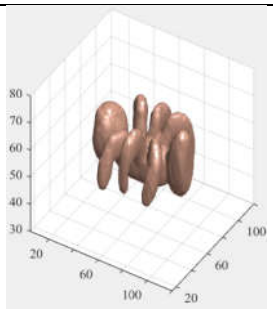
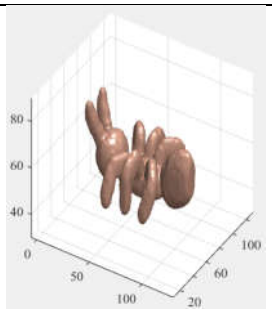
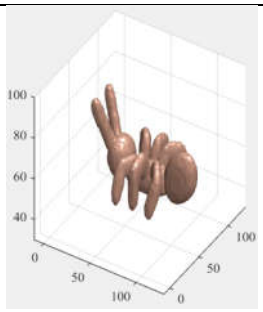
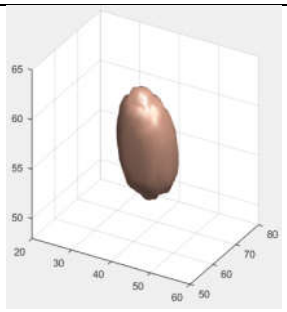
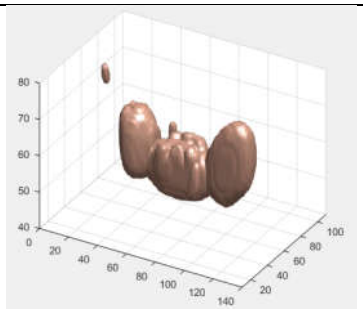
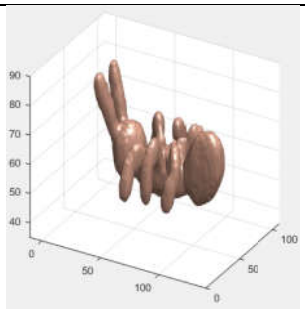
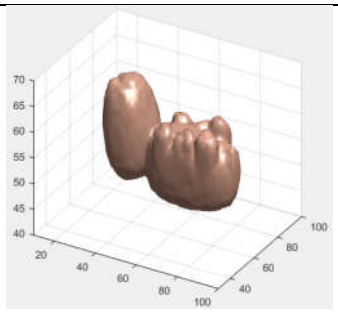
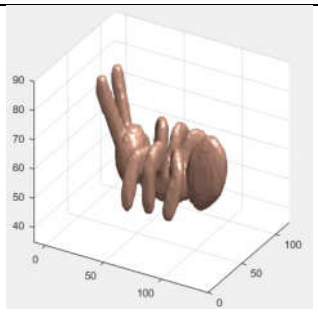
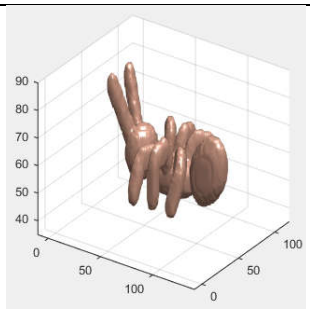
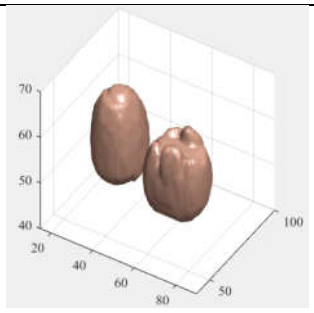
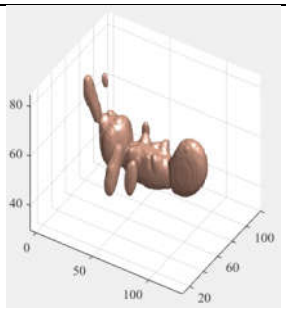
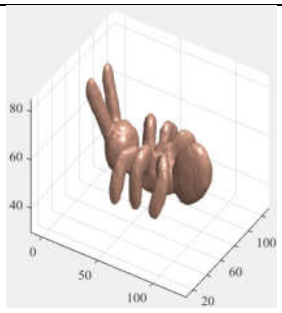
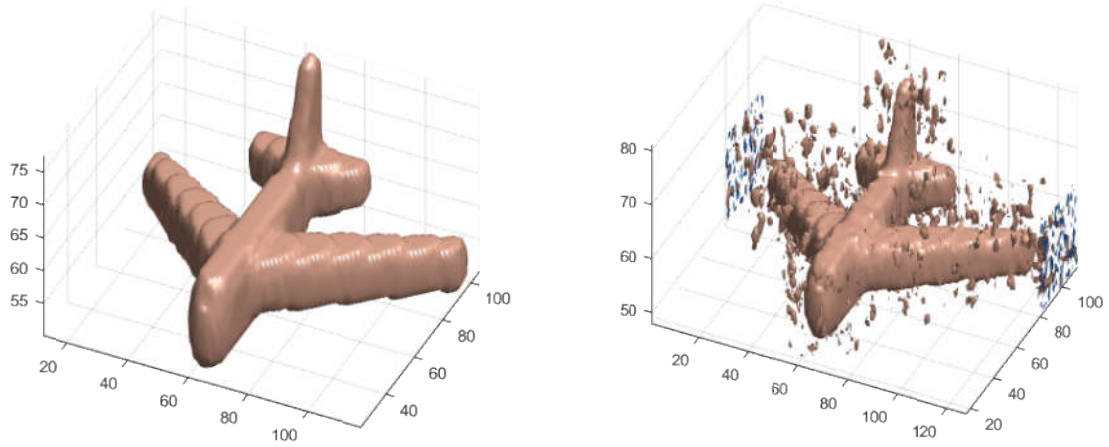
Order / Moments of	10	40	70
Tchebichef			
Krawtchouk			
Hahn			
Charlier			
Meixner			

Figure (IV.11): Image 3D reconstruite " Ant " pour les ordres 10, 40, 70,réspectivement par les moments de Tchebichef, Krawtchouk, Hahn, Charlier et Meixner.

Afin de tester la robustesse des moments de Meixner vis-à-vis du bruit avec la méthode proposée, nous allons utiliser un bruit aléatoire sur l'image «avion» (Figure IV.12) tiré de la base de données d'images McGILL benchMark [Mcgill, 2017].

La figure (IV.13) montre une comparaison des MSE des images reconstruites avec et sans bruit pour des ordres allant de 0 à 90. On peut voir que le MSE de l'image bruitée diminue avec la croissance des ordres, ce qui montre la robustesse des moments de Meixner par rapport au bruit aléatoire.



a. Noise-free "Airplane image" b. Noisy "Airplane image"

Figure (IV.12): Noise-free and noisy "Airplane image".

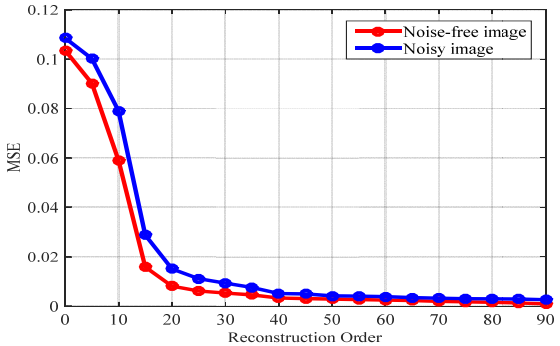


Figure (IV.13) Mean squared error « MSE » for the noise-free and noisy "Airplane image" by proposed method.

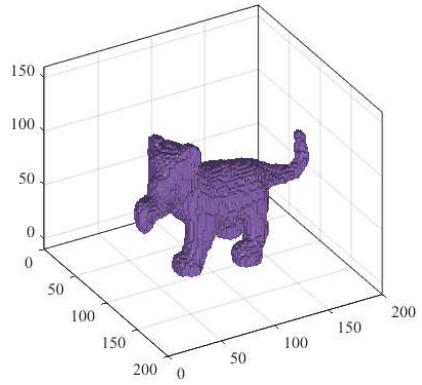
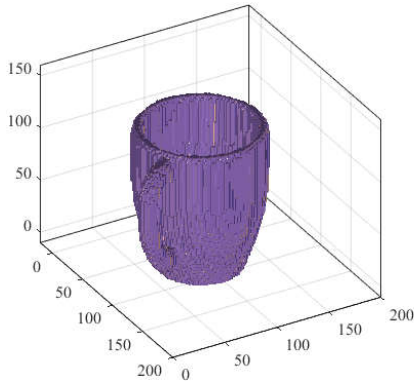
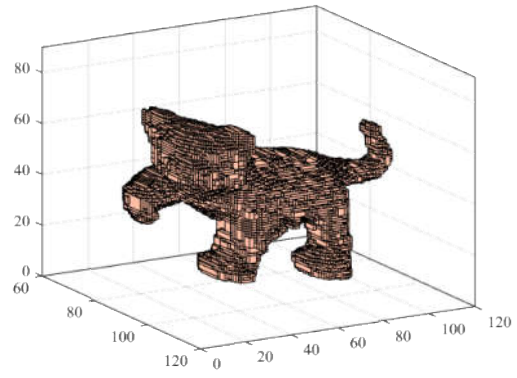
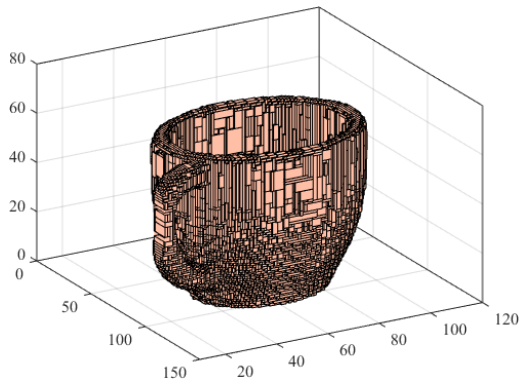
7. Simulations et résultats : Classification des mages 3D

Plusieurs expériences ont été menées pour évaluer la validité des moments invariants de Meixner proposés pour les images 3D.

7.1. Extraction des cuboïdes

Dans ce test, nous allons essayer d'extraire les cuboïdes d'une image 3D en utilisant l'algorithme ICR. En fait, nous allons considérer deux images binaires 3D “Dog” et “Cup” de taille $200\times200\times200$ voxels, extrait de la base “McGill 3D shape benchmark database (IV.14) et deux images médicales IRM en niveau de gris 3D “MRI 1” et “MRI 2” de taille $128\times128\times128$ voxels (IV.15). Les figures (IV.14 et IV.15) présentent l'ensemble des cuboïdes extraits de chaque image.

Les figures (IV.14 et IV.15) présentent le nombre des cuboïdes dans chaque image, on constate que le nombre des cuboïdes est beaucoup plus petit que la taille de l'image $200\times200\times200=8\ 000\ 000$ pour image binaire 3D et $128\times128\times128=2\ 097\ 152$ pour image multi-niveaux 3D, ce qui réduit considérablement le nombre de voxels traités. Cette réduction significative a un effet direct sur l'accélération du temps de calcul des moments invariants de Meixner, car ceux-ci dépendent du nombre de voxels dans les seins de chaque image ou de chaque cuboïde.

	Dog	Cup
Images originales 3D		
Représentation d'image par cuboïde (ICR)		
Nombre de cuboïde	2055	3741

Figue (IV.14): Représentation d'mage par cuboïde (ICR), pour des images binaires 3D " Dog " et " Cup "

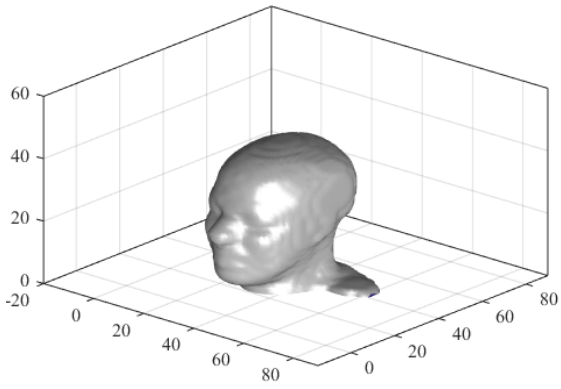
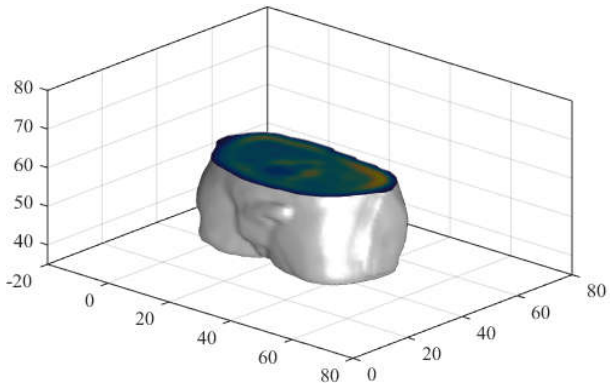
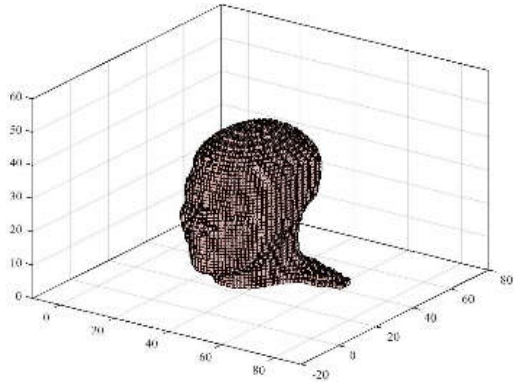
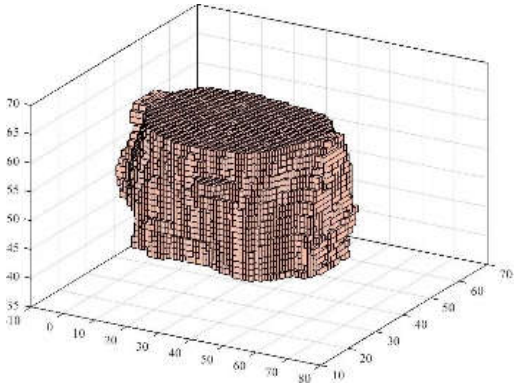
	MRI 1	MRI 2
Images originales 3D		
Représentation d'image par cuboïde (ICR)		
Nombre de cuboïde	37802	24307

Figure (IV.15): Représentation d'image 3D par cuboïde (ICR), pour des images multi-niveaux 3D " MRI 1" et " MRI 2"

7.2. Calcul des moments invariant orthogonaux de Meixner

Dans cette expérience, nous évaluons la vitesse de calcul des moments invariants de Meixner proposés, pour un ensemble de deux images binaires 3D “Dog” et “Cup” de taille $200\times200\times200$ voxels (IV.14), et pour un ensemble des deux images 3D en multi-niveaux “MRI 1” et ”MRI 2 ” de taille $128\times128\times128$ voxels (IV.15). Nous comparons d'abord le temps de calcul des moments invariants de Meixner par deux méthodes: la méthode proposée basée sur l'algorithme ICR et la méthode classique.

Le taux d'amélioration du temps d'exécution ETIR est utilisé comme critère [43] dans ce test. Ce ratio est défini comme $ETIR\ (\%) = (1 - Time1/Time2) \times 100$, où $Time1$ et $Time2$ sont le temps moyen de la première et de la deuxième, respectivement. Si ETIR=0 les deux temps d'exécution sont égaux.

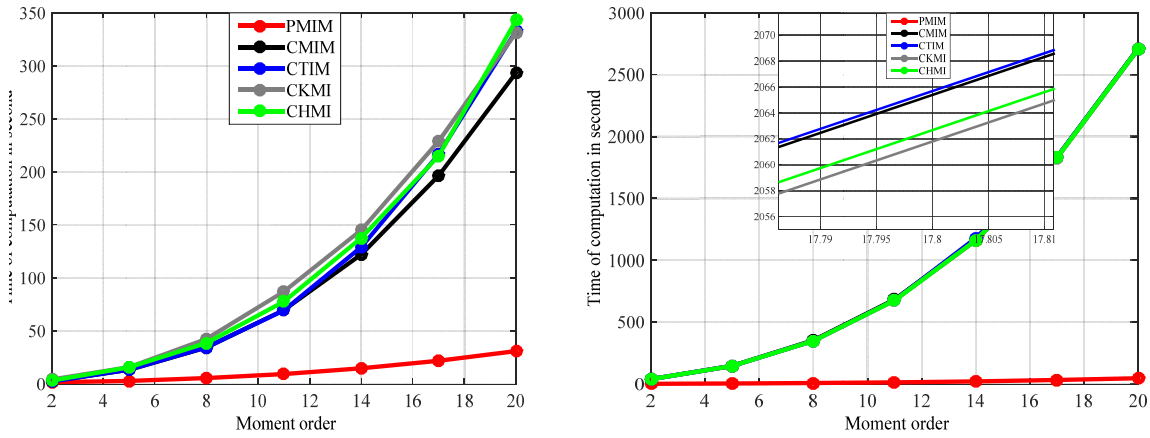
Les calculs sont effectués 15 fois pour chaque image et le temps moyen global requis pour calculer ces moments jusqu'à 20 ordres est présenté dans le tableau (IV.4). Ce tableau montre que la méthode proposée accélère considérablement le calcul des moments invariants. Notez que le ETIR le pourcentage pour les images binaires 3D est 88,05% et 98,13% pour les images multi-niveaux 3D, ce qui donne un pourcentage de réduction moyen proche de 93,09%.

Tableau (IV.4): Temps de calcul moyen global et ETIR (%) des moments invariants de Meixner selon la méthode proposée et classique pour les images binaires 3D et multi-niveaux.

Ensemble	**Calcul du temps moyen du MMI par**		
	Méthode proposée	Méthode Classique	ETIR(%)
Image binaire 3D	12.51	104.7	88.05
Image multi-niveaux 3D	18.42	989.5	98.13

Dans le second test, le temps de calcul des moments invariants de Meixner proposés (PMMI) est comparé aux moments Invariant de Meixner classiques (CMMI) [Sayyouri, 2015], aux moments Invariant de Tchebichef classiques (CTMI) [Batioua, 2017], aux moments Invariant de Krawtchouk classiques (CKMI). [Hmimid, 2015] et les moments Invariants de Hahn classiques (CHMI) [Sayyouri, 2013]. Les calculs sont effectués 15 fois pour chaque ensemble d'images 3D binaires et multi-niveaux et les temps moyen sont tracées sur la figure 3 en fonction de l'ordre des moments $(n+m+k)$ variant de 2 à 20 avec l'intervalle 3. Cette figure montre que nos moments invariants proposés de Meixner dans cet article vont trop vite vers les moments invariants de Meixner, Tchebichef, Krawtchouk et Hahn classiques, ce qui montre que notre méthode proposée réduit considérablement le temps de calcul des moments invariants à cause du calcul des moments invariants de Meixner par la méthode proposée en fonction du nombre de cuboïdes au lieu de la taille d'image 3D. Par

conséquent, nos descripteurs proposés seront souhaitables dans le domaine de la classification des images 3D. Notez que l'algorithme a été implémenté sur un PC Dual Core 2.10 GHz, 2 Go de RAM.

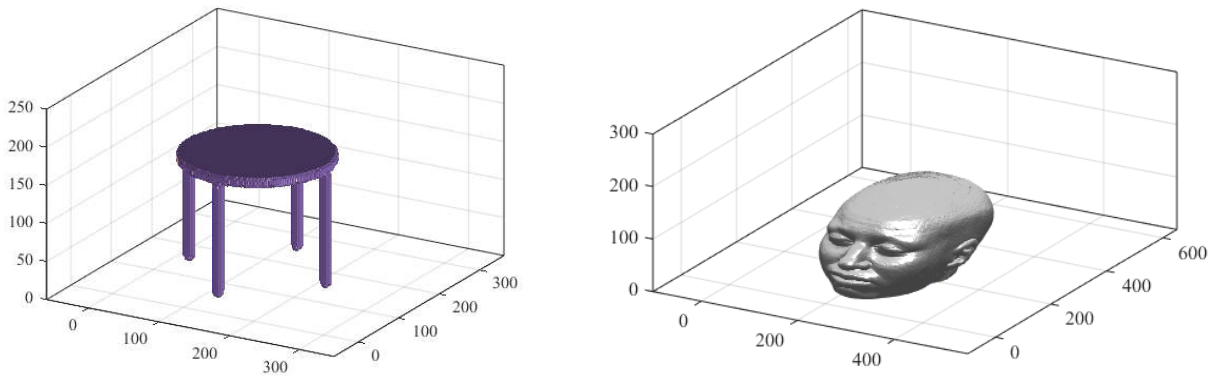


(a)

Figure (IV.16): Temps de calcul pour un ensemble d'images 3D, (a) : binaire, et (b) : multi-niveaux, en utilisant les moments invariant de : Meixner (CMIM), Tchebichef (CTIM), Krawtchouk (CKIM), Hahn classique (CHIM), et les moments invariants de Meixner proposés (PMIM).

7.3. Invariance TRS : Taranslation, Rotation, Scale

Dans cette expérience, nous allons tester l'invariance des moments invariants de Meixner proposés pour chaque transformation de translation, changement d'échelle et de rotation séparément dans le premier test et dans le second, l'invariance de ces moments est comparée à celle du classique pour les transformations mixtes.



(a) (b)

Figure (IV.17): L'image binaire 3D: (a) "Table", et l'image multi-niveaux 3D (b) "MRI 3" de taille 300×300×300 voxels.

Tout d'abord, nous vérifions l'invariance de translation, changement d'échelle et de rotation des moments invariants de Meixner proposés. Pour cela, nous utilisons les deux images 3D montrées dans (IV.17) de « Table » et « MRI 3 » de taille $300\times300\times300$ voxels. Ces images sont transformées par des facteurs changement d'échelle allant de 0,75 à 1,25 à pas de 0,05, transformées selon un vecteur de translation prenant des valeurs comprises entre (-16, -16, -16) et (16, 16, 16), et tourné d'un angle de rotation variant entre 0◦ et 360◦ avec un pas de 20. De plus, nous ajoutons du « bruit poivre et sel » avec différentes densités de bruit allant de 0 à 0,5 avec un pas de 0,05 afin de tester la robustesse de ces invariants pour les images dégradées.

L'invariabilité des moments invariants de Meixner proposés dans ce test est mesurée par l'erreur relative $E(I,\hat{I})$ entre les deux ensembles d'invariants correspondant à l'image d'origine $I(x,y)$ et l'image transformée $\hat{I}(x,y)$, qui défini comme [Hmimid, 2015] :

$$E(I,\hat{I})=\frac{\|\,IM(I)-IM(\hat{I})\,\|}{\|\,IM(I)\,\|} \qquad \textbf{(IV.66)}$$

où ‖:‖ est la norme euclidienne, les $IM(I)$ et $IM(\hat{I})$ sont les moments invariants pour l'image originale $I(x,y)$ et le transformé $\hat{I}(x,y)$, respectivement. Ces invariants du moment sont calculés jusqu'au 9ème ordre (n+ m+k ≤ 9). Les résultats sont représentés aux figures (IV.18, 19, 20 et 21).

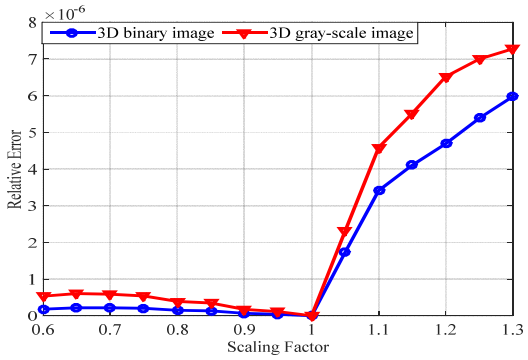


Figure (IV. 18) : Erreurs relatives pour l'image binaire 3D «Table» et l'image multi-niveaux 3D «MRI 3» entre les images originales et les versions à changement d'échelle en utilisant le PMMI proposé.

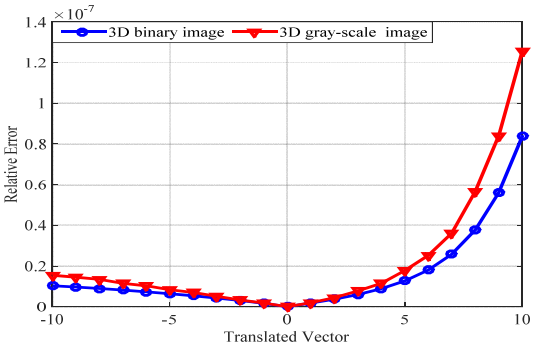


Figure (IV.19): Erreurs relatives pour l'image binaire 3D «Table» et l'image multi-niveaux 3D «MRI 3» entre les images originales et les versions à translation en utilisant le PMMI proposé.

Il ressort clairement de ces chiffres que les taux d'erreur relatifs sont très faibles, ce qui indique que notre méthode ICR proposée présente de bonnes performances et exprime une grande précision numérique sous différentes transformations géométriques, ainsi qu'en présence d'effets bruyants pour les deux images 3D.

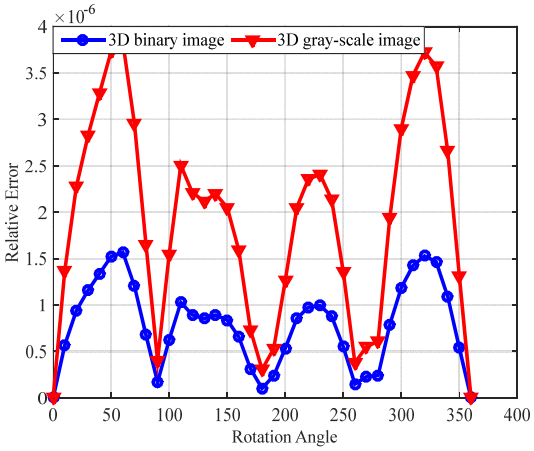


Figure (IV.20): Erreurs relatives pour l'image binaire 3D «Table» et l'image multi-niveaux 3D «MRI 3» entre les images originales et les versions à rotation en utilisant le PMMI proposé.

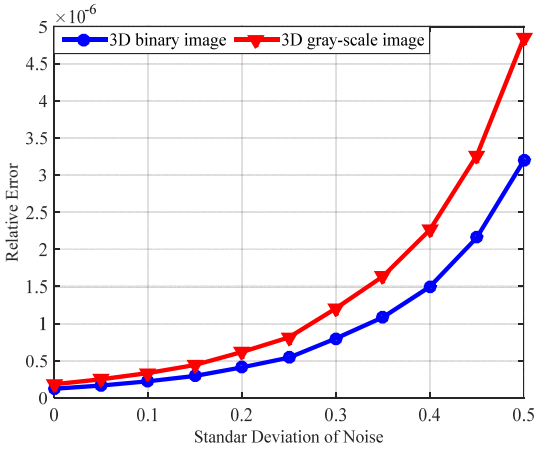


Figure (IV.21): Erreurs relatives pour l'image binaire 3D «Table» et l'image multi-niveaux 3D «MRI 3» entre les images originales et les versions à les images corrompues en utilisant le PMMI proposé.

Deuxièmement, nous comparons l'invariabilité des moments invariants de Meixner 3D par deux méthodes : la méthode ICR proposée et la méthode classique. Pour cela, les images ("Teddy-bears" et "MRI 3") dans (IV.22 et 23) sont transformés par un facteurs changement d'échelle s, vecteurs de translation $TV\left(T_x, T_y, T_z\right)$, et par une rotation des vecteurs $RV\left(\theta_x, \theta_y, \theta_z\right)$.



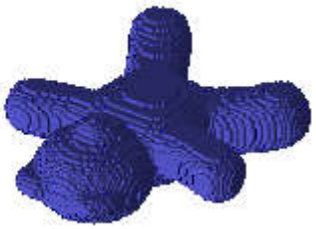




Image originale	Transformation 1: TV (-4, 4, 10); (θ_x=0°, θ_y=40°, θ_z=0°) s=0.7	Transformation 2: TV (4, 0, -5); (θ_x=10°, θ_y=10°, θ_z=105°) s=0.8







Transformation 3: TV (7, -1, 1); (θ_x=50°, θ_y=110°, θ_z=180°) s=0.9	Transformation 4: TV (10, -5, 0); (θ_x=180°, θ_y=30°, θ_z=10°) s=1.1	Transformation 5: TV (-3, -3, -3); (θ_x=70°, θ_y=130°, θ_z=20°) s=1.2

Figure (IV.22): Image transformée“Teddy-bears” utilisé dans cette expérience. L'image est montrée avec leurs transformations.

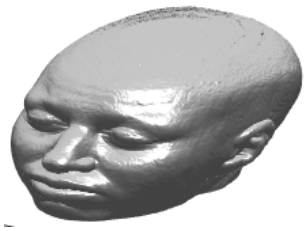
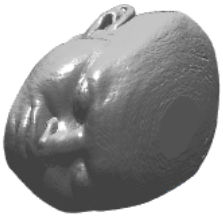
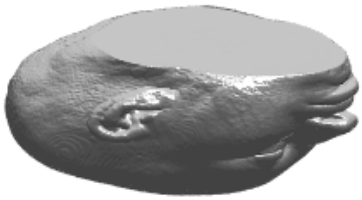
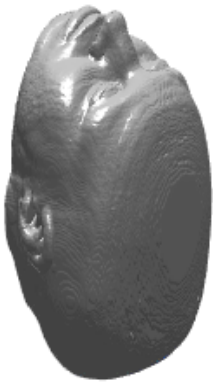




Image originale	Transformation 1: TV (-4, 4, 10); (θ_x=0°, θ_y=40°, θ_z=0°) s=0.7	Transformation 2: TV (4, 0, -5); (θ_x=10°, θ_y=10°, θ_z=105°) s=0.8

Transformation 3: TV (7, -1, 1); (θ_x=50°, θ_y=110°, θ_z=180°) s=0.9	Transformation 4: TV (10, -5, 0); (θ_x=180°, θ_y=30°, θ_z=10°) s=1.1	Transformation 5: TV (-3, -3, -3); (θ_x=70°, θ_y=130°, θ_z=20°) s=1.2

Figure (IV.23): Image transformée“ MRI 3 ” utilisé dans cette expérience. L'image est montrée avec leurs transformations.

Les figues Les figures (IV.22) et (IV.23) montrent les images transformées de "Teddy-bears" et de "MRI 3" utilisées dans ce test. La capacité de ces descripteurs sous diverses transformations géométriques est mesurée en termes de déviation des moments X. des invariants de Meixner, définie comme suit: $X = \sigma / |\mu|$ [Hmimid, 2015], où σ et μ désigne l'écart-type et la moyenne des moments invariants, respectivement.

Tableau (IV.5): Une comparaison des moments des invariants de Meixner 3D classiques et proposés d'image 3D "Teddy-bears".

Méthode	Transformation	MMI_{102}	MMI_{210}	MMI_{121}	MMI_{211}	MMI_{300}	MMI_{131}	MMI_{311}
Méthode proposée	Image Originale °	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5779E-26	1,4479E-25
	Transformation 1°	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5779E-26	1,4479E-25
	Transformation 2	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5779E-26	1,4479E-25
	Transformation 3	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5779E-26	1,4479E-25
	Transformation 4	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5779E-26	1,4479E-25
	Transformation 5	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5779E-26	1,4479E-25
	X=σ/|μ|(%)	**3,35E-13**	**1,53E-04**	**4,88E-05**	**6,88E-05**	**1,23E-04**	**9,70E-05**	**1,54E-04**
	Moyenne de *X(%)*	**9.22E-05**						
Méthode classique	Original image°	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5779E-26	1,4479E-25
	Transformation 1°	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5779E-26	1,4479E-25
	Transformation 2	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5779E-26	1,4479E-25
	Transformation 3	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5779E-26	1,4479E-25
	Transformation 4	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5780E-26	1,4479E-25
	Transformation 5	3,1028E-27	9,8133E-27	1,1918E-26	3,7694E-26	3,7694E-26	4,5779E-26	1,4479E-25
	X=σ/|μ|(%)	**5,81E-09**	**7,90E-03**	**1,002E-3**	**3,43E-3**	**3,11E-4**	**2,69E-4**	**6,28E-3**
	Moyenne de *X (%)*	**2,74E-3**						

Tableau (IV.6): Une comparaison des moments des invariants de Meixner 3D classiques et proposés d'image 3D "MRI 3".

Méthode	Transformation	MMI_{102}	MMI_{210}	MMI_{121}	MMI_{211}	MMI_{300}	MMI_{131}	MMI_{311}
Méthode proposée	Original image°	9,2663E-23	2,6755E-22	3,2438E-22	9,3660E-22	9,3660E-22	1,1356E-21	3,2787E-21
	Transformation 1°	9,2663E-23	2,6755E-22	3,2438E-22	9,3660E-22	9,3660E-22	1,1356E-21	3,2787E-21
	Transformation 2	9,2663E-23	2,6755E-22	3,2438E-22	9,3660E-22	9,3660E-22	1,1356E-21	3,2788E-21
	Transformation 3	9,2663E-23	2,6755E-22	3,2438E-22	9,3660E-22	9,3660E-22	1,1356E-21	3,2787E-21
	Transformation 4	9,2663E-23	2,6755E-22	3,2438E-22	9,3660E-22	9,3660E-22	1,1356E-21	3,2788E-21
	Transformation 5	9,2663E-23	2,6755E-22	3,2438E-22	9,3660E-22	9,3660E-22	1,1356E-21	3,2787E-21
	X=σ/|μ|(%)	**2,04E-13**	**1,32E-04**	**4,43E-05**	**1,60E-04**	**2,11E-04**	**1,06E-04**	**2,75E-04**
	Moyenne de *X (%)*	**1,33E-04**						
c e l a	Original image°	9,27E-23	2,68E-22	3,24E-22	9,37E-22	9,37E-22	1,14E-21	3,28E-21

	Transformation 1°	9,27E-23	2,68E-22	3,24E-22	9,37E-22	9,37E-22	1,14E-21	3,28E-21
	Transformation 2	9,27E-23	2,68E-22	3,24E-22	9,37E-22	9,37E-22	1,14E-21	3,28E-21
	Transformation 3	9,27E-23	2,68E-22	3,24E-22	9,37E-22	9,37E-22	1,14E-21	3,28E-21
	Transformation 4	9,27E-23	2,68E-22	3,24E-22	9,37E-22	9,37E-22	1,14E-21	3,28E-21
	Transformation 5	9,27E-23	2,68E-22	3,24E-22	9,37E-22	9,37E-22	1,14E-21	3,28E-21
	X=σ/|μ|(%)	**1,08E-10**	**1,35E-03**	**4,12E-03**	**1,19E-04**	**2,30162E-04**	**7,11E-03**	**1,17E-03**
	Moyenne de *X (%)*	**2,01E-03**						

Le pourcentage élevé de X représente la dispersion des valeurs par rapport à leurs moyennes correspondantes. Les tableaux (IV.5) et (IV.6) présentent les valeurs des moments d'invariants de Meixner d'ordre n, m, k ≤ 3 en utilisant la méthode proposée et la méthode classique pour les images “Teddy-bears” et “MRI 3”, respectivement.

Il ressort clairement de ces tableaux que les pourcentages $\sigma/|\mu|$ des moments invariants de Meixner proposés sont très faibles (10^{-6}), ce qui indique que ces moments présentent de bonnes performances et expriment une stabilité numérique élevée en transformations de translation, changement d'échelle et de rotation. Ces résultats montrent également la supériorité de la méthode proposée, qui donnent de meilleurs résultats que la méthode classique.

7.4. Classification

Enfin, pour évaluer l’efficacité des descripteurs proposés, nous testons l’exactitude de la reconnaissance de ces moments sur une base de données d’images 3D. Neuf images 3D (IV.24) de taille $128\times128\times128$, sont choisis parmi « McGill 3D Shape Benchmark database » [46-47], et utilisé pour la formation. Ainsi, afin de générer un ensemble de tests, ces images sont redimensionnées avec les facteurs 0.6, 0.8, 1.2 et 1.3, traduites par les vecteurs de translation. TV(-5, 5, 5), TV(3, -4, 1), TV(-4, 1, -6), et TV(2, -5, 7), et tourné par l'angle de rotation 30°, 80°, 130°, et 180° le long des axes x-,y-, et z, formant un ensemble de test de 576 images avec différentes transformations géométriques. Pour tester la robustesse de nos invariants contre l’effet négatif du bruit, le bruit poivre et sel est ajouté à l’essai de test avec différentes densités de bruit 1%, 2%, 3% et 4%.

Les voisins les plus proches (1-NN) basés sur la distance euclidienne d1 (IV.67), la distance de bloc de ville (City block distance) d2 (IV.68) et sur la distance de corrélation d3 (IV.69) sont utilisés comme classifieur dans cette expérience.

$$d_1(x_s, y_t^{(k)}) = \sqrt{\sum_{j=1}^{n} (x_{sj} - y_{tj})^2} \quad \textbf{(IV.67)}$$

$$d_2(x_s, y_t^{(k)}) = \sum_{j=1}^{n} | x_{sj} - y_{tj} | \quad \textbf{(IV.68)}$$

$$d_3(x_s, y_t^{(k)}) = 1 - \frac{\sum_{j=1}^{n} x_{sj} y_{tj}}{\sqrt{\sum_{j=1}^{n} (x_{sj})^2} \sqrt{\sum_{j=1}^{n} (y_{tj})^2}} \quad \textbf{(IV.69)}$$

Si les deux vecteurs xs et yt sont égaux, alors les distances d1, d2, et d3, tend vers 0.

où xs est le vecteur de caractéristique n-dimensionnelle d'image inconnue, et yt le vecteur de formation de classe k. Dans cette expérience, les vecteurs xs et yt sont calculés à l'ordre 3, et la précision de la classification est mesurée par la relation suivante :

$$\eta = \frac{\text{Nombre d'images correctement classées}}{\text{Le nombre total d'images utilisées dans le test}} \times 100\ \% \quad \textbf{(IV.70)}$$

La performance des moments invariants de Meixner proposés est vérifiée et comparée à celle des moments invariants de Meixner, Tchebichef, Krawtchouk et Hahn obtenus par la méthode classique. Les précisions de classification sont décrites dans les tableaux (IV.7, 8 et 9).

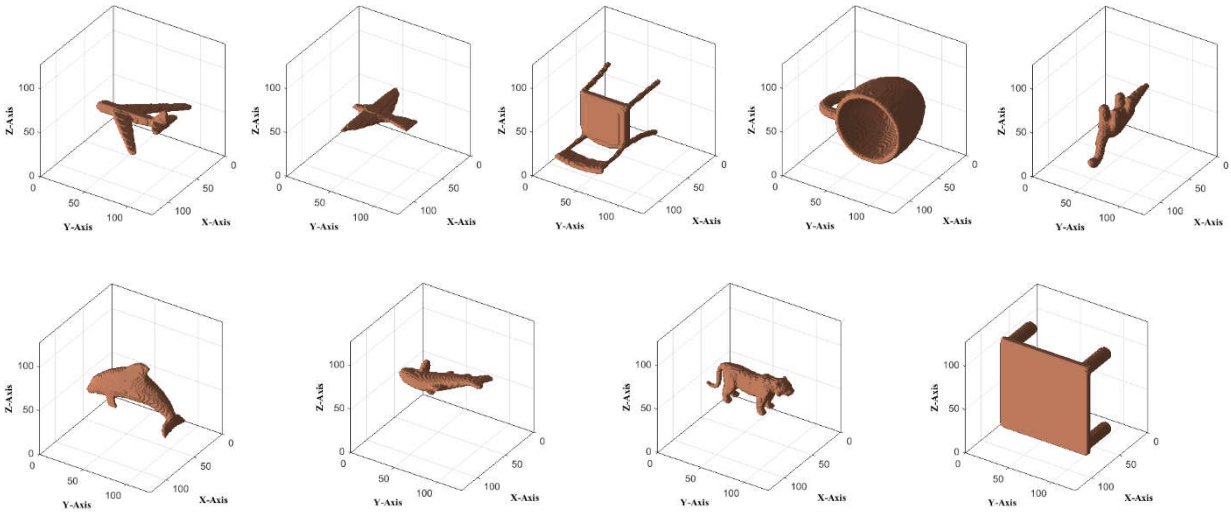


Figure (IV.24): Collection de 9 images sélectionnées dans la base de données PSB et utilisées comme ensemble d'apprentissage

Tableau (IV.7): Résultats de la classification utilisant le 1-NN basé sur la distance euclidienne d_1.

Moments Invariants	Sans bruit 0%	Bruit sel et poivre				**Moyenne**
		1%	2%	3%	4%	
PMMI	**86.46**	**81.75**	**79.70**	**77.85**	**75.22**	**80.20**
CMMI[36]	78.73	73.65	71.89	70.93	66.26	72.3
CTMI [27]	79.08	76.60	73.77	71.31	67.95	73.74
CKMI [31]	80.30	77.52	75.14	72.85	68.21	74.80
CHMI [32]	76.27	72.06	71.46	70.07	65.50	71.07

Tableau (IV.8): Résultats de classification utilisant 1-NN basé sur la distance des blocs de ville d2.

Moments Invariants	Sans bruit 0%	Bruit sel et poivre				**Moyenne**
		1%	2%	3%	4%	
PMMI	**86.45**	**81.62**	**79.78**	**77.98**	**75.22**	**80.21**
CMMI[36]	78.33	73.60	71.77	71.03	66.04	72.15
CTMI [27]	78.82	76.52	73.75	71.30	67.80	73.64
CKMI [31]	80.27	77.56	75.13	72.86	68.23	74.81
CHMI [32]	76.30	72.11	70.52	70.32	65.43	70.93

Tableau (IV.9): Résultats de classification utilisant le 1-NN basé sur la distance de corrélation d3.

Moments Invariants	Sans bruit 0%	Bruit sel et poivre				**Moyenne**
		1%	2%	3%	4%	
PMMI	**86.31**	**81.33**	**79.62**	**77.81**	**74.53**	**79.92**
CMMI[36]	78.20	73.47	71.04	70.66	65.82	71.84
CTMI [27]	78.84	75.91	73.02	70.57	67.46	73.16
CKMI [31]	79.60	77.15	74.72	72.68	67.75	74.38
CHMI [32]	75.63	71.80	71.00	69.53	65.22	70.66

À partir de ces tableaux, les moments invariants Meixner proposés 3D permettent une amélioration significative de la précision de la classification dans des conditions à la fois silencieuses et sans bruit. En comparaison avec les moments d'invariants classiques 3D, la précision de la classification s'est améliorée de 7,9%, 6,46%, 5,4% et 9,13% pour les moments d'invariants classiques de Meixner, Tchebichef, Krawtchouk et Hahn, respectivement, en utilisant le 1-NN basé sur la distance euclidienne. d_1, et amélioration presque identique en utilisant le 1-NN en fonction de la distance de bloc de ville d_2, et une amélioration de 8,08%, 6,76%, 5,54% et 9,26% respectivement pour ces moments, en utilisant le 1-NN en fonction de la distance de corrélation d_3. Ces résultats démontrent l'efficacité de la méthode proposée en termes de classification des images 3D.

8. Conclusion

À travers ce chapitre, nous avons essayé de proposer une nouvelle méthode de calcul des moments orthogonaux et de leurs inverses, pour réduire du temps de calcul énorme que consommé cette opération, mais aussi la réduction des erreurs de reconstruction.

Pour cela, nous avons choisi les polynômes de Meixner qui sont rarement étudiés, et nous avons proposé une nouvelle approche. La méthode proposée se compose de deux parties principales. La première est la nouvelle représentation d'images 3D ou de volumes discrets 3D en introduisant la représentation d'image par cuboïdes (ICR), inspirée des œuvres de Spiliotis et Papakostas sur les images 2D, ce qui nous a permis de réduire énormément le nombre d'élément à traiter, pour calculer les moments de l'image 3D. La deuxième, est

l’introduction du calcul matriciel pour booster plus le temps nécessaire pour calculer les moments orthogonaux et leurs inverses pour une image 3D.

Enfin, dans nos simulations, Nous avons réussi à prouver la rapidité indéniable de notre algorithme et son apport en termes d’amélioration de la reconstruction d’image 3D. Ainsi nous avons prouvé l’applicabilité de notre approche pour les différentes bases orthogonales et sa fiabilité. Pour valider la robustesse de notre approche, nous avons évalué son efficacité, à travers des simulations sur les images affectés par des bruits aléatoires.

CONCLUSIONS ET PERSPECTIVES

A travers ce travail de thèse, nous avons pu apporter quelques contributions à l'amélioration des algorithmes d'analyse d'images 2D et 3D par les moments orthogonaux discrets de Meixner. Ainsi, nous avons abordé quatre problématiques relevées dans la littérature, en proposant des solutions que nous avons expérimentées afin de prouver l'efficacité des approches et méthodes proposés.

La première problématique était le temps de calcul des moments orthogonaux discrets pour des images 2D, pour cette dernière nous avons adapté nos algorithmes une méthode numérique, notamment la récurrence de Clenshaw qui permet de réduire considérablement la complexité des calculs, ainsi en comparaison avec les méthodes classiques le temps de calcul devient nettement inférieure.

Ceci dit, il fallait aborder la deuxième problématique, celle de la stabilité numérique, pour le calcul des moments et pour la reconstruction d'image, ainsi nous avons introduit les filtres numériques pour assurer la stabilité numérique au niveau du calcul des moments orthogonaux discrets, et la répartition de l'image 2D en slices et blocs afin d'assurer la stabilité numérique au niveau de la reconstruction de l'image 2D. Notre approche a été validée à travers les tests effectués.

Le calcul des moments sur des images 3D, était la troisième problématique abordée, vu l'importance et l'évolution des images 3D, on ne pouvait que proposer une approche innovante pour la réduction du temps de calcul des images 3D. Ainsi nous avons introduit la représentation des images 3D par les cuboïdes (ICR), qui a permis de réduire le nombre d'élément à traiter sur une image 3D remarquablement et permettre de réduire le temps de calcul des moments pour des ordres importants à un temps proche du temps réel.

La dernière problématique était l'extraction des moments orthogonaux discrets pour la classification des images 3D, pour laquelle nous avons utilisé la représentation d'images

par cuboïdes, et prouvé à travers plusieurs tests l'amélioration de la classification des images 3D par les moments orthogonaux discrets.

Comme tout travail de recherche, notre modeste contribution s'inscrit dans la continuité des travaux initiés depuis un siècle et dont les débauchés sont des plus prometteuses. Cela dit, dans l'ère de l'intelligence artificielle et les données de masses, notre travail peut être amélioré au niveau de la classification et indexation des images des bases de données de grandes tailles, en y associant les algorithmes d'IA les plus adéquats, pour un réel apport dans le domaine des sciences et techniques appliquées, notamment pour l'imagerie médicale, pour les études géologiques, et dans l'industrie pour le contrôle de qualité au niveau 3D.

REFERENCES

[Abramowitz & Stegun, 1970] M. Abramowitz & I.A. Stegun (eds.) : Handbook of mathematical functions (with formulas, graphs, & mathematical tables). Dover Publications, New York, 1970.

[Aburdene, 1995] Aburdene, MF., Zheng, J., Kozick, RJ.: 'Computation of discrete cosine transform using Clenshaw's recurrence formula', IEEE Signal Processing Lett., 1995, vol. 2, pp. 155–156.

[Aquilanti et al., 2001] V. Aquilanti, S. Cavalli, C. Coletti, Angular & hyperangular momentum recoupling, harmonic superposition & Racah polynomial: a recursive algorithm, Chem. Phys. Lett. 344 (2001) 587–600.

[Belkasim et al., 1991] S.O. Belkasim, M. Shridhar, et M. Ahmadi. Pattern recognition with moment invariants : A comparative study & new results. Pattern Recognition, 24(12) :1117–1138, 1991.

[Chang, 2012] K.-H. Chang, R. Paramesran, B. Honarvar, and C.-L. Lim, "Efficient hardware accelerators for the computation of Tchebichef moments," IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 3, pp. 414–425, 2012.

[Chau, 1994] Chau, LP., Siu, WC.: 'Recursive algorithm for the discrete cosine transform with general lengths', Electron. Lett., 1994, vol. 30, no. 3, pp. 197–198.

[Chau, 2000] Chau, LP., Siu, WC.: 'Efficient recursive algorithm for the inverse discrete cosine transform', IEEE Signal Processing, 2000, Lett., vol. 7, pp. 276–277.

[Cherkaoui, 2009] Ajustement 3D par la méthode des moments développé pour une solution informatique à l'imagerie médicale IRM Cherkaoui, Abdeljabbar (Université Sidi Mohamed Ben Abdellah, Faculté des Sciences Dhar Mahraz, Fès, 2009-10-13)

[Chihara, 1978]T. S. Chihara: Introduction to Orthogonal Polynomials, Gordon & Breach, New York (1978).

[Chong et al., 2003] C.-W. Chong, P. Raveendran, & R. Mukundan, “Translation invariants of Zernike moments,” Pattern Recogn. 36, 1765–1773 (2003).

[Chong et al., 2004] C.-W. Chong, P. Raveendran, & R. Mukundan, “Translation & scale invariants of Legendre moments,” Pattern Recogn. 37, 119–129 (2004).

[Deng et al., 2009] Deng C, Gao X, Li X, Tao D (2009), A local Tchebichef Moments-based robust image watermarking. Signal Process 89:1531–1539.

[Fadili, 2003] El Fadili, H., Zenkouar, K., Qjidaa, H.: Lapped Block Image Analysis Via the Method of Legendre Moments. EURASIP Journal on Applied Signal Processing 2003(9), 902–913 (2003)

[Flusser & Suk, 1993] Flusser, J., Suk, T., 1993. Pattern recognition by affine moment invariants. Pattern Recognition 26, 1, pp. 167-174.

[Flusser, 2016] J. Flusser, T. Suk, B. Zitova, 2D and 3D image analysis by moments. John Wiley & Sons, 2016.

[Fu, 2008] B. Fu, J.-Z. Zhou, Y.-H. Li, B. Peng, L.-Y. Liu, and J.-Q. Wen, “Novel recursive and symmetric algorithm of computing two kinds of orthogonal radial moments,” Image Science Journal, vol. 56, no. 6, pp. 333–341, 2008.

[Hatamian, 1986] M. Hatamian, A real-time two-dimensional moment generating algorithm and its single chip implementation, IEEE Trans. Acoust., Speech Signal Process. 34 (3) (1986) 546–553.

[Hilbert, 1993] D. Hilbert, Theory of Algebraic Invariants. Cambridge University Press, 1993.

[Hmimd et al., 2013]A. Hmimd, M. Sayyouri , H. Qjidaa, “A Fast Computation of Continuous Orthogonal Moments for Binary Gray-Scale Images ”. 1er Workshop sur l'Imagerie, Systèmes et Applications - ISA'2013 - , 23-24 Mai 2013 ,Taza, Maroc.

[Hmimid et al., 2012] Hmimid, A.; Sayyouri, M.; Qjidaa, H., "A fast method for reconstruction of binary & gray-scale images by the tchebichef moments," Information Science & Technology (CIST), 2012 Colloquium in , vol., no., pp.106,111, 22-24 Oct. 2012.

[Hmimid et al., 2014] A. Hmimid ; M. Sayyouri and H. Qjidaa "Image classification using a new set of separable two-dimensional discrete orthogonal invariant moments", J. Electron Imaging. 23(1), 013026 (Feb 18, 2014). ; http://dx.doi.org/10.1117/1.JEI.23.1.013026.

[Hmimid, 2015] A Hmimid, Mhamed Sayyouri, and Hassan Qjidaa, Fast computation of separable two-dimensional discrete invariant moments for image classification, Pattern Recognition 48 (2015) 509–521.

[Honarvar, 2013] B. Honarvar, R. Paramesran, C.-L. Lim, The fast recursive computation of Tchebichef moment and its inverse transform based on Z-transform, Digital Signal Process. 23 (5) (2013) 1738–1746.

[Honarvar, 2014a] B. Honarvar, R. Paramesran, A new formulation of geometric moments from lower output values of digital filters, J. Circ., Syst. Comput. (2014) 1450055.

[Honarvar, 2014b] B. Honarvar, J. Flusser, Fast computation of Krawtchouk moments, Information Sciences 288 (20) (2014) 73–86

[Hosny, 2011] K.M. Hosny, Fast and low-complexity method for exact computation of 3D Legendre moments. Pattern Recognition Letters, 2011, vol. 32, no 9, p. 1305-1314.

[Hosny, 2011] Khalid M. Hosny, Image representation using accurate orthogonal Gegenbauer moments, Pattern Recognition Letters, Volume 32, Issue 6, 15 April 2011, Pages 795-804.

[Hosny, 2012] Khalid M. Hosny, Fast computation of accurate Gaussian-Hermite moments for image processing applications, Digital Signal Processing, v.22 n.3, p.476-485, May, 2012

[Hu, 1962] M. K. Hu, “Visual pattern recognition by moment invariants,” IRE Trans. Inform. Theory 8, 179–187 (1962).

[Hwang, 2006] S.-K. Hwang and W.-Y. Kim, “A novel approach to the fast computation of Zernike moments,” Pattern Recognition, vol. 39, no. 11, pp. 2065–2076, 2006.

[Jahid, 2017] T. Jahid, A. Hmimid, H. Karmouni, M. Sayyouri, H. Qjidaa and A. Rezzouk. Image analysis by Meixner moments and a digital filter. Multimedia Tools and Applications, 2017, p. 1-21.

[Jahid, 2018] T. Jahid, H. Karmouni, M. Sayyouri et al. Fast Algorithm of 3D Discrete Image Orthogonal Moments Computation Based on 3D Cuboid. Journal of Mathematical Imaging and Vision, 2018, p. 1-21.

[Jahid, 2018a] T. Jahid, A. Hmimid, H. Karmouni, M. Sayyouri, H. Qjidaa and A. Rezzouk. Image analysis by Meixner moments and a digital filter. Multimedia Tools and Applications, 2018, vol. 77, no 15, p. 19811-19831.

[Karakasis et al., 2013] E.G. Karakasis, G.A. Papakostas, D.E. Koulouriotis & V.D. Tourassis, “Generalized Dual Hahn Moment Invariants”, Pattern Recognition, vol. 46, no. 7, pp. 1998-2014, 2013.

[Karmouni, 2018]H. Karmouni, A. Hmimid, T. Jahid, M. Sayyouri, H. Qjidaa, A. Rezzouk, "Fast and Stable Computation of the Charlier Moments and Their Inverses Using Digital Filters and Image Block Representation." Circuits, Systems, and Signal Processing: 1-19. (2018)

[Karmouni, 2019] H. Karmouni, T. Jahid, M. Sayyouri et al. Fast Reconstruction of 3D Images Using Charlier Discrete Orthogonal Moments. Circuits, Systems, and Signal Processing, 2019, p. 1-28.

[Koekoek & Swarttouw, 1998] R. Koekoek & R. Swarttouw: The Askey-scheme of hypergeometric orthogonal polynomials & its q-analogue. Report no. 98-17, (1998), Faculty of Information Technology & Systems, Delft University of Technology.

[Koekoek et al., 2010] R. Koekoek, P. A. Lesky, & R. F. Swarttouw, Hypergeometric Orthogonal Polynomials & Their q-Analogues, Springer Monographs in Mathematics, Library of Congress Control Number: 2010923797 (Springer-Verlag, 2010).

[Kotoulas & Andreadis, 2005] L. Kotoulas et I. Andreadis. Real-time computation of Zernike moments. IEEE Transactions on Circuits & Systems for Video Technology, 15(6) :801–809, June 2005.

[Liao & Pawlak, 1996] Liao, S.X., Pawlak, M., 1996. On image analysis by moments. IEEE Transactions on Pattern Analysis & Machine Intelligence 18, 3, pp. 254-266.

[McGill, 2017] 3D image Database http://www.cim.mcgill.ca/~shape/benchMark/airplane.html. (2017). Accessed 31 July 2017

[MPEG-7, 2014] MPEG-7 DATA SET, http://www.dabi.temple.edu/shape/MPEG7/dataset.html.

[Mukundan & Ramakrishnan, 1998] Mukundan, R., Ramakrishnan, K.R., 1998. Moment functions in image analysis Theory & applications. World scientific publishing Co. Pte. Ltd., Singapore.

[Mukundan et al., 2001] R.Mukundan, , S. H. Ong, & P. A. Lee, "Image analysis by Tchebichef moments," IEEE Trans. Image Processing, vol. 10, no. 9, pp. 1357-1364, 2001.

[Mukundan, 2004] Mukundan, R. (2004) "Some computational aspects of discrete orthonormal moments," IEEE Transactions on Image Processing, vol. 13, no. 8, pp. 1055–9.

[Nikiforov & Uvarov, 1988] A.F. Nikiforov, V.B. Uvarov: Special functions of Mathematical Physics, Birkhauser,Basel, Boston, (1988).

[Nikiforov et al., 1991] A. F. Nikiforov, S. K. Suslov, & V. B. Uvarov, Classical Orthogonal Polynomials of a Discrete Variable (Springer,1991).

[Nikolajevic, 2003] Nikolajevic, V., Fettweis, G.: 'Computation of forward and inverse MDCT using Clenshaw's recurrence formula', IEEE Trans Signal Processing, 2003, vol. 51, pp. 1439–1444.

[Papakostas et al., 2008] G.A. Papakostas, E.G. Karakasis & D.E. Koulouriotis, "Efficient & Accurate Computation of Geometric Moments on Gray-Scale Images", Pattern Recognition, vol. 41, no. 6, pp. 1895-1904, 2008.

[Papakostas et al., 2010a]G.A. Papakostas, E.G. Karakasis & D.E. Koulouriotis, "Novel Moment Invariants for Improved Classification Performance in Computer Vision Applications", Pattern Recognition, vol. 43, no. 1, pp. 58-68, 2010.

[Papakostas et al., 2010b]G.A. Papakostas, E.G. Karakasis & D.E. Koulouriotis, “Accurate & Speedy Computation of Image Legendre Moments for Computer Vision Applications”, Image & Vision Computing, vol. 28, no. 3, pp. 414-423, 2010.

[Papakostas et al., 2010c]G.A. Papakostas, D.E. Koulouriotis, E.G. Karakasis, Computation strategies of orthogonal image moments: A comparative study, Applied Mathematics & Computation, Volume 216, Issue 1, 1 March 2010.

[Papakostas, 2010d] G.A. Papakostas, E.G. Karakasis & D.E. Koulouriotis, “Novel Moment Invariantss for Improved Classification Performance in Computer Vision Applications”, Pattern Recognition, vol. 43, no. 1, pp. 58-68, 2010.

[Preoţiuc-Pietro, 2017] Preoţiuc-Pietro, D., Liu, Y., Hopkins, D., & Ungar, L. (2017). Beyond binary labels: political ideology prediction of Twitter users. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Vol. 1, pp. 729-740).

[Sadjadi & Hall, 1980]Sadjadi, F.A., Hall, E.L., 1980. Three-Dimensional Moment Invariants. IEEE Transactions on Pattern Analysis & Machine Intelligence PAMI-2(2), 2, pp. 127-136.

[Sadjadi, 1978] F. A. Sadjadi and E. L. Hall, "Numerical computation of moment invariants for scene analysis," in Proc. IEEE Conf: on Pattern Recognition and Image Processing, Chicago, IL, 1978.

[Sayyouri et al., 2011] M. SAYYOURI , A. HMIMD, H. QJIDAA, “A 3d image reconstruction by the method of discrete orthogonal moments” . 3éme édition des Journées Doctorales en Technologies de l’Information et de la Communication: JDTIC 7, 8 & 9 Juillet 2011 Tanger.

[Sayyouri et al., 2012a] M. Sayyouri, A. Hmimd, & H. Qjidaa, “A fast computation of Hahn moments for binary & gray-scale images,” in IEEE International Conference on Complex Systems (ICCS’12), Agadir, Morocco, November 5–6, 2012.

[Sayyouri et al., 2012b] M. Sayyouri, A. Hmimd, & H. Qjidaa, “A fast computation of Charlier moments for binary & gray-scale images,” in The 2nd Edition of the IEEE

Colloquium on Information Sciences & Technology (CIST'12), Fez, Morocco, October 22–24, 2012.

[Sayyouri et al., 2012c] M. SAYYOURI ,A. HMIMD, H. QJIDAA, " A fast computation of charlier & meixner moments for binary & gray-scale images ".Euro-Mediterranean Conference on Bioengineering & Biomaterials EMCBB-2012, July, 4, 5, 6, 2012 Fes Maroc.

[Sayyouri et al., 2012d] M. SAYYOURI ,A. HMIMD, H. QJIDAA, " A fast computation of krawchouk moments for binary & gray-scale images ".International Conference on Software Engineering, Databases & Expert Systems (SEDEXS'12) 14,15 & 16 juin 2012 Settat, Maroc.

[Sayyouri et al., 2012e]M. SAYYOURI ,A. HMIMD, H. QJIDAA, "A fast method for reconstruction of binary & gray-scale images by pollaczek's moments". 4èmes Journées Doctorales en Technologies de l'Information et de la Communication (JDTIC'12) ,Casablanca du 05 au 07 juillet 2012,Maroc.

[Sayyouri et al., 2013a] M. Sayyouri, A. Hmimid, & H. Qjidaa, "Improving the performance of image classification by Hahn moment invariants," J. Opt. Soc. Am. A (JOSA A) 30, 2381-2394 (2013).

[Sayyouri et al., 2013b]M. Sayyouri, A. Hmimd, H. Qjidaa, "A fast computation of discrete orthogonal moments for binary gray-scale images ". 1er Workshop sur l'Imagerie, Systèmes et Applications - ISA'2013 - , 23-24 Mai 2013 ,Taza, Maroc.

[Sayyouri et al., 2014a] M. Sayyouri, A. Hmimd, H. Qjidaa, "Fast method for image reconstruction by Hahn's discrete orthogonal moments". The first International Workshop on Wireless Technologies and Distributed Systems- WITS'2014 , 9-10 Avril 2014 ,Fès, Maroc.

[Sayyouri, 2015] M. Sayyouri, A. Hmimid, H. Qjidaa, A fast computation of novel set of Meixner invariants moments for image analysis. Circuits Syst. Signal Process. 34(3), 875–900 (2015)

[Sayyouri, 2016] M. Sayyouri, A. Hmimid, H. Qjidaa, Image analysis using separable discrete moments of Charlier– Hahn. Multimed. Tools Appl. 75(1), 547–571 (2016)

[Sayyouri, 2014] thèse :Contributions en analyse d'image par la méthodes des moments orthogonaux discrets (Université Sidi Mohamed Ben Abdellah, Faculté des Sciences Dhar Mahraz, Fès, 2014)

[Shen, 1997] Shen, J., 1997. Orthogonal Gaussian-Hermite moments for image characterization, SPIE Intelligent Robots Computer Vision XVI, Pittsburgh, 15 Oct., pp. 224-233.

[Sheng & Shen, 1994] Sheng, Y. & Shen, L. (1994) "Orthogonal Fourier-Mellin moments for invariant pattern recognition," Journal of the Optical Society of America A, vol. 11, no. 6, pp. 1748–57.

[Shu et al., 2010] H. Z. Shu, H. Zhang, B. J. Chen, P. Haigron, & L. M. Luo, "Fast computation of Tchebichef moments for binary & gray-scale images," IEEE Trans. Image Process. 19, 3171–3180 (2010).

[Shu, 2007] Shu, HZ. Luo, LM. Coatrieux, JL. "Moment-based appraoches in image Part1: basic feautures.", IEEE, Eng Med Biol Mag. 2007;26(5) pp70-75.

[Spiliotis & Mertzios,1998] I. M. Spiliotis & B. G. Mertzios, "Real-time computation of two-dimensional moments on binary images using image bloc representation," IEEE Trans. Image Process. 7, 1609–1615 (1998).

[Suslov, 1989] S.K. Suslov: The theory of difference analogues of special functions of hypergeometric type, Russian Math, Surveys, 44 (1989), 227-278.

[Szego, 1959] Szego, G. Orthogonal Polynomials. American Mathematical Society Colloquium Publications, Vol. 23. Revised ed. American Mathematical Society, Providence, R.I. 1959.

[Teague, 1980] Teague, M.R., 1980. Image analysis via the general theory of moments. Jour. of Optical Society America 70, pp. 920-930.

[Teh & Chin, 1988] Teh, C.H., Chin, R.T., 1988. On Image Analysis by the Methods of Moments. IEEE Transactions on Pattern Analysis & Machine Intelligence 10, 4, pp. 496-513.

[Upneja, 2015] R. Upneja and C. Singh, “Fast computation of Jacobi-Fourier moments for invariants image recognition,” Pattern Recognition, vol. 48, no. 5, pp. 1836–1843, 2015.

[Wallin & Kübler, 1995] Wallin, A. & Kübler, O. (1995) “Complete sets of complex Zernikemoment invariants & the role of the pseudoinvariants,” IEEE Transactions Pattern Analysis & Machine Intelligence, vol. 17, no. 11, pp. 1106–10.

[Wang, 1994] Wang, Z., Jullien, GA., Miller, WC.: 'Recursive algorithms for the forward and inverse discrete cosine transform with arbitrary length', IEEE Signal Processing Lett., 1994, vol. 1, pp. 101–102.

[Wang, 2004] Wang, GB., Wang, SG.: 'Parallel recursive computation of the inverse Legendre moment transforms for signal and image reconstruction', IEEE Signal Processing Lett., 2004, vol.11,no.12,pp.929-932.

[Wang, 2006] Wang, GB., Wang, SG.: 'Recursive Computation of Tchebichef Moment and its inverse transform', Pattern Recognition, 2006, vol.39,pp.47-56.

[Wang, 2014] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli. (2004). Image quality assessment: From error visibility to structural similarity, IEEE Transactions on Image Processing 13(4): 600–612, (2014).

[Wang,2009] Z. Wang, A. C. Bovik. Mean squared error: Love it or leave it? A new look at signal fidelity measures, IEEE Signal Processing Magazine 26(1): 98–117, (2009).

[Wee & Paramesran, 2006] C-Y. Wee & R. Paramesran. Efficient computation of radial moment functions using symmetrical property. Pattern Recogn., 39(11) :2036–2046, 2006.

[Wee & Paramesran, 2007] C-Y.Wee et R. Paramesran. On the computational aspects of Zernike moments. Image Vision Comput., 25(6) :967–980, 2007.

[Wee, 2007] C.-Y. Wee and R. Paramesran, “Derivation of blur-invariant features using orthogonal Legendre moments,” IET Computer Vision, vol. 1, no. 2, pp. 66–77, 2007.

[Wong, 1999] W.-H. Wong, W.-C. Siu, Improved digital filter structure for fast moments computation, IEE Proc. – Vis., Image Signal Process. 146 (2) (1999) 73–79.

[Wu & Shen, 2005] Wu, Y. & Shen, J. (2005) “Properties of orthogonal Gaussian–Hermite moments & their applications,” EURASIP Journal on Applied Signal Processing, no. 4, pp. 588–99.

[Wu, 2013] H. Wu, J. L. Coatrieux, and H. Shu, “New algorithm for constructing and computing scale invariants of 3D Tchebichef moments,” Math. Probl. Eng. 2013, 813606 (2013).

[Yap et al., 2003] P.T. Yap, R. Paramesran, S.H. Ong, Image analysis by Krawtchouk moments IEEE Trans Image Processing 12 (2003) 1367–1377.

[Yap et al., 2007] P. T. Yap, P. Raveendran, & S. H. Ong, “Image analysis using Hahn moments,” IEEE Trans. Pattern Anal. Mach. Intell. 29, 2057–2062 (2007).

[Zhang et al, 2010] H. Zhang, H.Z. Shu, P. Haigron, B.S. Li, L.M. Luo, Construction of a complete set of orthogonal Fourier–Mellin moment invariants for pattern recognition applications, Image & Vision Computing, Volume 28, Issue 1, January 2010, Pages 38-44

[Zhang, 2010a] H. Zhang, H. Shu, G.-N. Han, G. Coatrieux, L. Luo, and J. L. Coatrieux, “Blurred image recognition by Legendre moment invariants,” IEEE Transactions on Image Processing, vol. 19, no. 3, pp. 596–611, 2010.

[Zhi, 2018] R. Zhi, L. Cao, et G. Cao. Translation and scale invariantss of Krawtchouk moments. Information Processing Letters, 2018, vol. 130, p. 30-35.

[Zhu et al, 2007a] Zhu, H.Q., Shu, H.Z., Liang, J., Luo, L.M., Coatrieux, J.L., 2007a. Image analysis by discrete orthogonal Racah moments. Signal Processing 87, 4, pp. 687-708.

[Zhu et al, 2007b]H. Zhu, H. Shu, T. Xia, L. Luo, & J. L. Coatrieux, “Translation & scale invariants of Tchebichef moments,” Pattern Recogn. 40, 2530–2542 (2007).

[Zhu et al, 2007c]H. Zhu, H. Shu, J. Zhou, L. Luo, & J. L. Coatrieux, “Image analysis by discrete orthogonal dual Hahn moments,” Pattern Recogn. Lett. 28, 1688–1704 (2007).

[Zhu et al, 2010] Zhu, H.; Liu, M.; Shu, H.; Zhang, H.; Luo, L., "General form for obtaining discrete orthogonal moments," Image Processing, IET , vol.4, no.5, pp.335,352, October 2010. Applied Mathematics & Computation 189, 2, pp. 1214-1222.