

Table des matières

Remerciements	1
Introduction	9
1 Théories des Moments	12
1.1 Introduction	12
1.2 Définition de Moment	12
1.3 Les moments invariants de Hu	12
1.4 Les moments Orthogonaux Discrets	13
1.4.1 Le Moment de Tchebichef	14
1.4.2 Les Moments de Krawtchouk	17
1.4.3 Le Moment de Hahn	22
1.4.4 Le Moment de DualHahn	25
1.5 Conclusion	28
2 Reconstruction des Images 2-D à partir des Moments Discrets	30
2.1 Introduction	30
2.2 Étude Expérimentale	31
2.2.1 Méthode	31
2.2.2 Reconstruction des images 2-D à partir de Moments Tchebichef . . .	31
2.2.3 Reconstruction des images 2-D à partir des Moments de Krawtchouk	37
2.2.4 Reconstruction des images 2-D à partir de Moments Hahn	42
2.2.5 Reconstruction des images 2-D à partir de Moments de DualHahn . .	47
2.2.6 Comparaison générale	53
2.3 Conclusion :	54
3 Application à la reconnaissance de Forme :	55

3.1	Introduction	55
3.2	Méthode de Classification Supervisée	55
3.2.1	K-plus proche voisin	57
3.2.2	Le Perceptron Multicouche	59
3.3	Résultats et Analyse	63
3.3.1	Extraction des caractéristiques	63
3.3.2	Reconnaissance des chiffres manuscrits - BD MNIST -	64
3.3.3	Reconnaissance de visage - BD YALE FACE -	65
3.3.4	Analyse générale et discussion	67
3.4	Conclusion	67
4	Application de reconstruction des Images 2-D en niveau de gris via les moments discrets	68
4.1	Introduction	68
4.2	Logiciel de Développement	68
4.3	Réalisation et contraintes	69
4.4	Modélisation	69
4.5	Application Réalisée	70
4.6	Conclusion	73
	Conclusion	74

Table des figures

1.1	le tracé de polynôme de Tchebichef normalisé pour $N=20$	16
1.2	la propriété de symétrie pour les polynômes de Tchebichef	17
1.3	le tracé de polynôme de Krawtchouk normalisé et pondéré-normalisé pour $p = 0.5$ et $N = 100$	20
1.4	Comparaison entre les polynômes de Krawtchouk pondérés-normalisés pour $p = 0.3$ et $p = 0.7$	20
1.5	le tracé de polynôme de Krawtchouk pondéré-normalisé pour $N=100$ et $p=0.5$	22
1.6	le tracé de polynôme de Hahn normalisé pour $N=60$, $\nu = 0$ et $\mu = 0$	23
1.7	le tracé de polynôme de DualHahn normalisé pour $N=b=40$, $a=c=0$	27
1.8	l'influence de paramètre -a- pour les polynômes normalisés de DualHahn avec $b = N+a$, $c=a$ et (a) $a = 0$, (b) $a=-5$, (c) $a = 12$	29
1.9	l'influence de paramètre -c- pour les polynômes normalisés de DualHahn, $a = 8$, $b = 48$. (a) $c = -8$, (b) $c = -4$, (c) $c = 0$, (d) $c = 4$, (e) $c = 8$	29
2.1	image binaire de chiffre 7 et lettre E	33
2.2	reconstruction des images binaires pour l'ordre $n+m=8$, $n+m=20$, $n+m=28$, $n+m=32$ et $n+m=38$	33
2.3	la courbe de MSE pour différents ordres de reconstruction du chiffre -7- et la lettre -E-	34
2.4	la courbe de PSNR pour différents ordres de reconstruction du chiffre -7- et la lettre -E-	34
2.5	image d'expérimentation, Cameramman, Lena, Papper, de taille $64*64$	34
2.6	Reconstruction des images niveau de gris pour différents ordres	34
2.7	la courbe de MSE pour différents ordres de reconstruction	35
2.8	la courbe de PSNR pour différents ordres de reconstruction	35
2.9	la reconstruction des images bruitées avec bruit impulsional de densité $d=0.02$	35
2.10	la reconstruction des images bruitées avec bruit gaussienne de densité $d=0.02$	36

2.11 le taux d'erreur de reconstruction des images bruitées avec bruit impulsional de densité $d=0.02$	36
2.12 le taux d'erreur de reconstruction des images bruitées avec bruit gaussien de densité $d=0.02$	36
2.13 Extraction des caractéristiques pour l'ordre $n+m=60$	38
2.14 reconstruction des images binaires pour ordre $n+m=8$, $n+m=20$, $n+m=28$, $n+m=32$ et $n+m=38$	39
2.15 la courbe de MSE pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-	39
2.16 la courbe de PSNR pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-	39
2.17 reconstruction des image niveau de gris pour ordre $n+m=20$, $n+m=30$, $n+m=50$, $n+m=70$, $n+m=100$ et $n+m=124$	40
2.18 la courbe MSE pour différents ordres de reconstruction	40
2.19 la courbe PSNR pour différents ordres de reconstruction	40
2.20 la reconstruction des images bruitées avec bruit gaussienne de densité $d=0.02$	41
2.21 la reconstruction des images bruitées avec bruit impulsional de densité $d=0.02$	41
2.22 le taux d'erreur de reconstruction des reconstruction des images bruitées avec bruit gaussienne de densité $d=0.02$	42
2.23 taux d'erreur de reconstruction des reconstruction des images bruitées avec bruit impulsional de densité $d=0.02$	42
2.24 reconstruction des images binaires pour ordre $n+m=8$, $n+m=20$, $n+m=28$, $n+m=32$ et $n+m=38$	44
2.25 la courbe de MSE pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-	44
2.26 la courbe de PSNR pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-	44
2.27 reconstruction des images niveau de gris pour ordre $n+m=20$, $n+m=30$, $n+m=50$, $n+m=70$, $n+m=100$ et $n+m=124$	45
2.28 courbe MSE pour différents ordres	45
2.29 courbe PSNR pour différents ordres	45
2.30 reconstruction des images bruité avec bruit impulsional de densité $d=0.02$	46
2.31 reconstruction des images bruité avec bruit gaussien de densité $d=0.02$	46
2.32 reconstruction des images bruitées avec bruit impulsional de densité $d=0.02$	47
2.33 reconstruction des images bruitées avec bruit gaussienne de densité $d=0.02$	47

2.34	reconstruction des images binaires pour les ordres $n+m=8$, $n+m=20$, $n+m=28$, $n+m=32$ et $n+m=38$	49
2.35	la courbe de MSE pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-	50
2.36	la courbe de PSNR pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-	50
2.37	reconstruction des images niveau de gris pour ordre $n+m=20$, $n+m=30$, $n+m=50$, $n+m=60$, $n+m=80$ et $n+m=94$	50
2.38	courbe MSE pour différents ordres	51
2.39	courbe PSNR pour différents ordres	51
2.40	reconstruction des images bruitées avec bruit gaussienne de densité $d=0.02$. .	51
2.41	reconstruction des images bruitées avec bruit impulsionnel de densité $d=0.02$.	52
2.42	reconstruction des images bruitées avec bruit impulsionnel de densité $d=0.02$.	52
2.43	reconstruction des images bruité avec bruit gaussienne de densité $d=0.02$. . .	52
2.44	courbe MSE pour différents ordres de l'image Cameraman	53
2.45	courbe PSNR pour différents ordres de l'image Cameraman	53
2.46	courbe MSE pour différents ordres de l'image Lena	53
2.47	courbe PSNR pour différents ordres de l'image Lena	53
2.48	courbe MSE pour différents ordres de l'image Papper	54
2.49	courbe PSNR pour différents ordres de l'image Papper	54
3.1	système de classification	56
3.2	Exemple d'un réseau de type perceptron multicouche	59
3.3	Modèle de neurone j	61
3.4	Exemple d'apprentissage	64
3.5	Exemple de test	64
3.6	Taux d'apprentissage pour chaque moment	65
3.7	Taux d'apprentissage d'algorithme de rétropropagation, 10 couches cachées .	65
3.8	Taux d'apprentissage d'algorithme de rétropropagation, 20 couches cachées .	65
3.9	Exemple d'apprentissage	66
3.10	Exemple de test	66
3.11	Taux d'apprentissage pour chaque moment	66
3.12	Taux d'apprentissage d'algorithme de rétropropagation, 10 couches cachées .	67
3.13	Taux d'apprentissage d'algorithme de rétropropagation, 20 couches cachées .	67

4.1	diagramme de cas d'utilisation	69
4.2	Diagramme de séquence	70
4.3	accueil de l'application	71
4.4	Reconstruction de Lena via le moment de Tchebichef	71
4.5	reconstruction de Pepper via le moment de Krawtchouk, ajout de bruit gaussien	72
4.6	reconstruction de Cameramman via le moment de Hahn, ajout de bruit im- pulsionnel	72
4.7	reconstruction du chiffre 7 via le moment de DualHahn	73

Introduction Générale

La théorie des moments est largement utilisée dans le domaine de traitement d'image à savoir : analyse d'images[14][12][17], la détection de contour [18][19] et la reconnaissance de forme [15][16]. L'idée fondamentale de la théorie des moments est la projection de l'espace des données sur une base souvent orthogonale pour extraire les informations utiles, cette dernière est extraite sous forme des coefficients de projection appelés moments.

La notion de théorie des moments était introduite par M. K. Hu [1] en 1962, ce dernier a généré un ensemble de moments géométriques invariants. En effet, Hu a présenté en [1] sept moments invariants par rapport aux différentes déformations géométriques (rotation, échelle et translation), permettant ainsi l'extraction des caractéristiques globales d'une image. Ensuite, Y. S. Abu-Mostafa [11] a utilisé les moments de Hu dans le domaine de reconnaissance de forme. Cependant l'inconvénient des moments de Hu est que la base de projection n'est pas orthogonale, impliquant ainsi une redondance d'information[13].

Pour surmonter le problème, Teague en 1980 [14] a introduit les moments orthogonaux calculés à partir des polynômes orthogonaux. Dans cette étude, Teague était le premier à utiliser les moments pour la reconstruction d'image en exploitant les propriétés de non redondance et de réversibilité qui sont dues à l'orthogonalité de la base utilisée. Ensuite, Teh et Chin exposent dans [12] une étude détaillée sur différents types de moments dont : géométrique, Legendre, Zernike, Pseudo-Zernike, Une comparaison a ensuite été établie en terme de qualité de représentation, redondance de l'information et sensibilité au bruit, avec des résultats analytiques et expérimentaux. La conclusion tirée par les auteurs lors de cette étude est que les moments orthogonaux sont bien meilleurs que les autres moments en terme de qualité de reconstruction et redondance d'information.

Cependant, ces moments orthogonaux (Legendre, Zernike, Pseudo-Zernike ...) impliquent généralement plusieurs problèmes du fait qu'ils utilisent un espace continu sachant que l'image est discrète, citons comme exemple : l'approximation numériques des intégrales continues, transformation des coordonnées (passage de l'espace continue à l'espace discret) et les coûts de calcul élevés et finalement la détermination de l'ordre optimal de reconstruction [7].

C'est pour cette raison dans ce mémoire, nous optons pour l'utilisation des moments

orthogonaux discrets car ils détiennent la plupart des fonctionnalités utiles dans les moments orthogonaux continus. En plus, l'implémentation des moments orthogonaux discrets ne nécessite aucune approximation numérique, du moment que la base polynomial est orthogonale discrète dans l'espace. Historiquement, en 2001 Mukundun [3] est parmi les pionniers de l'utilisation des moments discrets (Le moment de Tchebichef) en analyse d'image surtout en reconstruction. Depuis, leurs utilisations en traitement d'image n'a cessé d'accroître [5-7].

Le présent travail est divisé en deux axes majeurs :

- Le premier axe a pour objectif d'établir une étude comparative en termes de pouvoir de reconstruction entre les différents types des moments orthogonaux discrets à savoir : Tchebichef, Krawtchouk, Hahn, DualHahn. Dans cette étude, le PSNR (Peak Signal to Noise Error) sera utilisé comme critère objectif pour mesurer la qualité d'image reconstruite.
- D'autre part, le second axe de ce travail a pour finalité de concevoir un système de reconnaissance d'image en utilisant les moments discrets orthogonaux de Tchebichef, Krawtchouk et Hahn comme descripteurs de forme. Notre système de classification proposé contient deux principaux modules : module d'extraction des attributs et module de classification. L'extraction des vecteurs attributs moments est réalisée en trois phases. Premièrement, le calcul des moments, ensuite la détermination de l'ordre optimal en utilisant le critère MSE. Enfin la phase d'extraction du vecteur attributs pertinents pour chaque image de la base de données. Lors de l'étape de classification deux algorithmes différents seront proposés le KNN (K-plus proche voisins) et ANN (Réseau de neurones). La capacité de discrimination de notre système de classification sera testée sur différentes bases de données hétérogènes, MNIST [20] qui concerne la reconnaissance des chiffres manuscrits et YALE FACE [21] pour la reconnaissance de visage, s'en suivra une étude comparative entre les différents types de moments en terme de pouvoir de discrimination et de redondance d'information.

En résumé les travaux réalisés au niveau de ce mémoire portent sur une présentation générale de la théorie des moments discrets, en particulier les moments de Tchebichef, Krawtchouk, Hahn et DualHahn. Ensuite, une partie sera consacrée au pouvoir de reconstruction de ces moments traités. Nous proposerons un système de classification qui se base sur les moments comme attributs d'entrée au classifieur. Finalement, une application sera implémentée pour présenter le pouvoir de reconstruction proposé pour chacun des moments traités.

Ce présent mémoire est organisé en quatre chapitres de la manière suivante :

- Dans le premier chapitre, nous introduisons les concepts de la théorie des moments orthogonaux discrets. Dans ce cadre, nous présenterons en détails les moments de Tchebi-

chef, Krawtchouk, Hahn et DualHahn.

- Le deuxième chapitre est consacré à l'étude du pouvoir de reconstruction concernant les moments orthogonaux discrets proposés dans le premier chapitre, on utilisera pour la comparaison les images niveau de gris et binaires.
- Dans le troisième chapitre, l'état de l'art de la classification supervisée, sera présentée. En particulier les deux algorithmes, KNN et Réseaux de Neurones. Ensuite nous proposons un système de classification qui utilise les moments comme attributs. Les deux bases de données hétérogènes MNIST et YALE FACE permettant d'évaluer les performances de chaque moment en termes de pouvoir de discrimination.
- Le dernier chapitre est consacré à la présentation d'un module de l'application réalisée durant notre travail. Cette application permettra l'implémentation de pouvoir de reconstruction des images 2-D avec une évaluation des taux d'erreur obtenus pour chacun des moments proposés dans notre travail.

Enfin la conclusion générale présente une synthèse de travail réalisé, ainsi que les pistes définissant des perspectives possibles pour les travaux futurs.

Chapitre 1

Théories des Moments

1.1 Introduction

Dans ce chapitre, nous proposons une étude détaillée concernant la théorie des moments discrets dans leur sens mathématique, dont le but est de connaître la théorie fondamentale, concernant le calcul des moments et la reconstruction des images 2-D (niveau de gris et binaire). Nous allons se focaliser sur quatre types les plus connus des moments orthogonaux discrets, qui sont, Tchebichef, Krawtchouk, hahn et DualHahn.

1.2 Définition de Moment

Les moments sont des grandeurs scalaires utilisées pour caractériser une fonction et extraire ses propriétés importantes.

De point de vue physique, les moments sont introduits dans la discipline de mécanique classique de corps rigide pour mesurer la distribution de masse d'un corps.

Du point de vue mathématiques les moments sont introduits dans la discipline de statistiques et probabilité, pour l'ordre $n=2$ le moment signifie la variance d'une série statistique et pour $n=1$ signifie l'écart type, d'une façon générale les moments signifiant une projection d'une fonction sur une base polynomiale.

1.3 Les moments invariants de Hu

Hu en 1962 [1] est le premier qui a introduit la notion des théories des moments en traitement d'image, là où il a présenté sept moments invariants, afin de les exploiter en domaine de reconnaissance de forme, et qui sont indépendants au déformation géométrique (échelle, translation, rotation).

En dimension 2-D le moment géométrique d'ordre (n+m) est donné par :

$$m_{nm} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^n y^m f(x, y) dx dy \quad \text{avec } p, q = 0, 1, 2, \dots \quad (1.1)$$

Il faut noter qu'au niveau d'équation (1.1) les moments géométriques obtenus n'assurent pas les différentes variations géométriques (translation, l'échelle et rotation). Les moments invariants seront obtenus en utilisant la notion des moments centraux que l'on définit comme suit :

$$\mu_{nm} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x - \bar{x})^n (y - \bar{y})^m f(x, y) dx dy \quad \text{avec } p, q = 0, 1, 2, \dots \quad (1.2)$$

tel que les deux points \bar{x} et \bar{y} sont le centre de l'image donnée par : $\bar{x} = \frac{m_{10}}{m_{00}}$ et $\bar{y} = \frac{m_{01}}{m_{00}}$.

D'après (1.2) les moments calculés sont invariants par rapport à la translation, pour pallier le problème de l'échelle et la rotation, les moments peuvent être obtenus par une normalisation des moments centraux, Hu l'on définit comme suit :

$$\eta_{nm} = \frac{\mu_{nm}}{\mu_{00}^\lambda} \quad \text{avec } \lambda = (n + m + 1)/2, \quad p, q = 0, 1, 2, \dots \quad (1.3)$$

D'après l'équation (1.3) les sept moments sont donnés comme suit :

$$\begin{aligned} \phi_1 &= \eta_{20} + \eta_{02}. \\ \phi_2 &= (\eta_{20} + \eta_{02})^2 + 4\eta_{11}^2. \\ \phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2. \\ \phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} - \eta_{03})^2. \\ \phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ &+ (3\eta_{21} - \eta_{03})(\eta_{21} - \eta_{03})[3(\eta_{30} - \eta_{12})^2 - (\eta_{21} - \eta_{03})^2]. \\ \phi_6 &= (\eta_{20} - \eta_{02})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ &+ 4\eta_{11}(\eta_{30} - \eta_{12})(\eta_{21} - \eta_{03}). \\ \phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ &- (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]. \end{aligned}$$

Ces sept moments restent suffisants pour que l'information soit invariable par rapport aux différents déformations géométriques.

1.4 Les moments Orthogonaux Discrets

Les moments orthogonaux constituent des quantités statistiques obtenues par la projection d'une image dans une base polynomiale orthogonale, à cause de leurs orthogonalités, ils offrent la possibilité d'extraction des caractéristiques d'une image dont le but de pouvoir les utiliser au domaine de reconnaissance des formes, compression de données et la reconstruction des images[2].

Les moments orthogonaux discrets sont introduits pour pallier le problème des erreurs d'approximation engendrées par les moments orthogonaux continus (Zernike, Legendre,...),

et qui sont causées par le passage des formules continues en formule discrètes[6], dans les sections qui suivent nous présentons les moments discrets de Tchebichef, Krawtchouk, Hahn, et finalement DualHahn.

1.4.1 Le Moment de Tchebichef

Mukundan et al ont présenté dans [3] leur approche de calcul des moments en se basant sur les polynômes discrets de Tchebichef. Le polynôme de Tchebichef d'ordre n est donné comme suit [3] :

$$t_n(x) = (1 - N)_n {}_3F_2(-n, -x, 1 + n; 1, 1 - N; 1) \quad (1.4)$$

$$t_n(x) = (1 - N)_n \sum_{k=0}^n \left(\frac{(-n)_k (-x)_k (1 + n)_k}{(k!)^2 (1 - N)_k} \right) \quad (1.5)$$

avec $n, x = 0, 1, 2, \dots, N$

Notez que ${}_3F_2$ représente la fonction hypergéométrique généralisée définie par :

$${}_3F_2(a_1, a_2, a_3; b_1, b_2; z) = \sum_{k=0}^{\infty} \left(\frac{(a_1)_k (a_2)_k (a_3)_k z^k}{(k!) (b_1)_k (b_2)_k} \right) \quad (1.6)$$

étant donné que a_k représente le symbole de Pochhammer défini par :

$$(a)_k = a(a + 1)(a + 2) \dots (a + k - 1), \quad k \geq 1 \quad \text{et} \quad a_0 = 1 \quad (1.7)$$

pour une image de taille $N \times N$ les polynômes de Tchebichef satisfont le critère d'orthogonalité :

$$\sum_{x=0}^{\infty} t_m(x) t_n(x) = \rho(n, N) \delta_{nm}, \quad 0 \leq m, n \leq N \quad (1.8)$$

sachant que δ_{nm} est le symbole de Kroniker et $\rho(n, N)$ représente la norme quadratique donnée par :

$$\rho(n, N) = \frac{N(N^2 - 1) \dots (N^2 - 2^2)(N^2 - n^2)}{2n + 1} \quad n = 0, 1, \dots, N - 1 \quad (1.9)$$

Afin d'éviter l'instabilité numérique donnée par la fonction hypergéométrique au niveau de la formule (1.5), en [3] Ils ont présenté la relation de récurrence pour les polynômes de Tchebichef discrets classiques, la relation est comme suit :

$$(n + 1)t_{n+1}(x) - (2n + 1)(2x - N + 1)t_n(x) + n(N^2 - n^2)t_{n-1}(x), \quad n = 1, 2, \dots, N - 1 \quad (1.10)$$

le moment de Tchebichef est donné par la relation suivante :

$$T_{nm} = \frac{1}{\rho(n, N)\rho(m, N)} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} t_n(x) t_m(x) f(x, y) \quad n, m = 0, 1, 2, \dots, N - 1 \quad (1.11)$$

Dans les deux relations (1.10) et (1.11) avec un simple calcul on peut vérifier que les valeurs de t_n et T_{nm} augmentent au valeur de N^n et $N^{-(p+q)}$ successivement, ce qui implique un taux d'erreur de calcul assez important. Afin de corriger ce problème, en [3] ils ont introduit les polynômes de Tchebichef normalisés, qui sont donnés par :

$$\bar{t}_n(x) = \frac{t_n(x)}{\beta(n, N)} \quad (1.12)$$

avec $t_n(x)$ représente le polynôme de Tchebichef classique obtenu par la formule (1.10) et $\beta(n, N) = N^n$ représente une suite constante indépendante de x , de même la norme quadratique sera modifiée de la façon suivante :

$$\bar{\rho}(n, N) = \frac{\rho(n, N)}{\beta(n, N)^2} \quad (1.13)$$

$$\bar{\rho}(n, N) = \frac{N(1 - \frac{1}{N^2})(1 - \frac{2^2}{N^2}) \dots (1 - \frac{n^2}{N^2})}{2n + 1}, \quad \text{avec } n = 0, 1, \dots, N - 1 \quad (1.14)$$

Les polynômes de Tchebichef normalisés sont donnés par la formule récurrente suivante :

$$\bar{t}_n(x) = \frac{(2n - 1)\bar{t}_1(x)\bar{t}_{n-1}(x) - (n - 1)(1 - \frac{(n-1)^2}{N^2})\bar{t}_{n-2}(x)}{n} \quad (1.15)$$

$$\bar{t}_0(x) = 1 \quad (1.16)$$

$$\bar{t}_1(x) = \frac{2x + 1 - N}{N} \quad (1.17)$$

Après avoir présenté les polynômes de Tchebichef normalisés, le moment de Tchebichef sera modifié et calculé à partir de nouveaux polynômes obtenus par la formule (1.15), tandis que la formule de moment modifiée est :

$$T_{nm} = \frac{1}{\bar{\rho}(n, N)\bar{\rho}(m, N)} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \bar{t}_n(x)\bar{t}_m(x)f(x, y) \quad n, m = 0, 1, 2, \dots, N - 1 \quad (1.18)$$

Remarque :

Nous avons énoncé au début que les polynômes de Tchebichef sont orthogonaux, or cette orthogonalité n'est validée que dans l'intervalle $[-1, 1]$, ce qui porte une condition nécessaire au cours de calcul des moments de Tchebichef. Afin de montrer cette remarque la figure (1.1) représente le tracé de six courbes, chacune représente le polynôme de Tchebichef normalisé d'ordre $n = (0, 1, 2, 3, 4, 5)$.

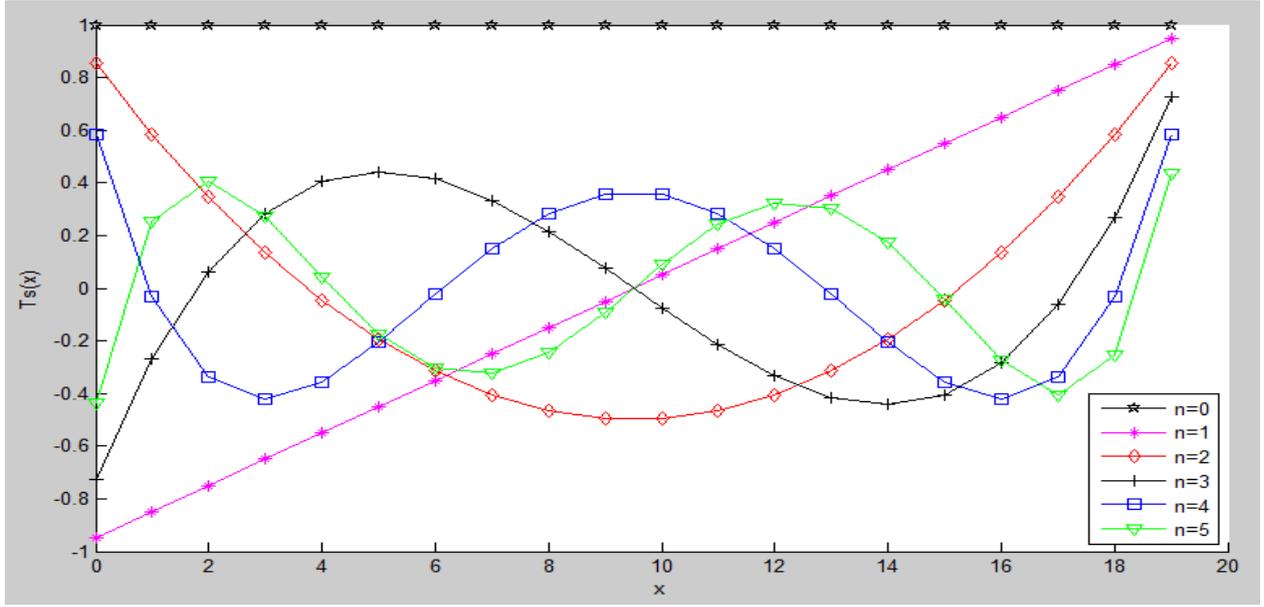


FIGURE 1.1 – le tracé de polynôme de Tchebichef normalisé pour $N=20$.

Après avoir présenté le pouvoir de compression des images 2-D avec les moments de Tchebichef, on peut même présenter le pouvoir de reconstruction à partir des moments calculé en (1.18), la formule inverse de reconstruction est donnée par :

$$f(x, y) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} \bar{t}_n(x) \bar{t}_m(y) T_{mn} \quad x, y = 0, 1, 2, \dots, N-1 \quad (1.19)$$

Afin d'améliorer la complexité de programme soit au niveau de calcul de moment ou au niveau de reconstruction, les polynômes de Tchebichef normalisés ont les mêmes propriétés des polynômes classiques qui satisfont la propriété de symétrie donnée par :

$$\bar{t}_n(N-1-x) = (-1)^n \bar{t}_n(x) \quad (1.20)$$

Cette relation permet de subdiviser une image de dimension $N \times N$ en quatre parties égaux (Figure 1.2), ce qui va influencer au niveau de la complexité de l'algorithme, car on aurait besoin de calculer que le premier quart $0 \leq x, y \leq (N/2 - 1)$, donc l'expression de calcul de moment donné par la formule (1.18) sera modifiée comme suit :

$$T_{nm} = \frac{1}{\bar{\rho}(n, N) \bar{\rho}(m, N)} \sum_{x=0}^{(N/2)-1} \sum_{y=0}^{(N/2)-1} \bar{t}_n(x) \bar{t}_m(y) \left\{ \begin{array}{l} f(x, y) + (-1)^n f(N-1-x, y) \\ + (-1)^m f(x, N-1-y) \\ + (-1)^{n+m} f(N-1-x, N-1-y) \end{array} \right\} \quad (1.21)$$

Comme valeur ajoutée de la propriété de symétrie, elle permet de minimiser le stockage des polynômes de Tchebichef, si une application a besoin de calculer le max de degré M pour une image 2-D de taille $N \times N$, dans ce cas une matrice de polynôme de taille $M \times N/2$

sera suffisante, d'autre part la formule de reconstruction (1.19) sera modifiée de la façon suivante :

$$\begin{aligned}
 f(x, y) &= \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} \bar{t}_n(x) \bar{t}_m(x) T_{mn} && \text{si } x, y < (N/2) \\
 f(x, y) &= \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} (-1)^m \bar{t}_m(N-1-x) \bar{t}_n(y) T_{mn} && \text{si } x \geq (N/2); y < (N/2) \\
 f(x, y) &= \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} (-1)^n \bar{t}_m(x) \bar{t}_n(N-1-y) T_{mn} && \text{si } x < (N/2); y \geq (N/2) \\
 f(x, y) &= \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} (-1)^{n+m} \bar{t}_m(N-1-x) \bar{t}_n(N-1-y) T_{mn} && \text{si } x, y \geq (N/2)
 \end{aligned}
 \tag{1.22}$$

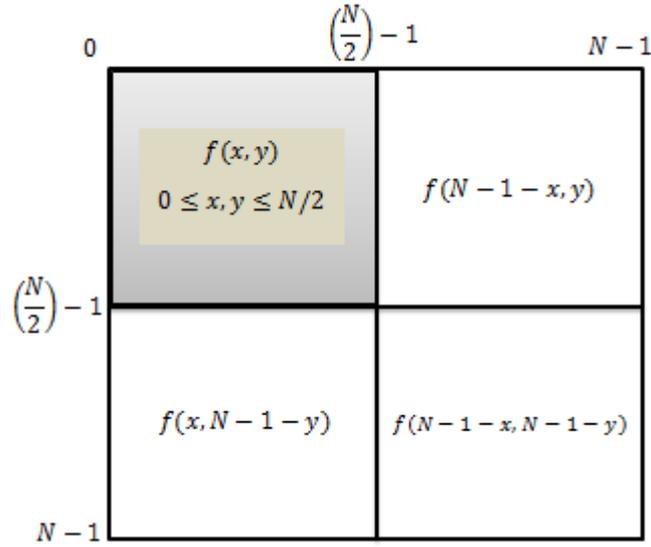


FIGURE 1.2 – la propriété de symétrie pour les polynômes de Tchebihef

Après avoir présenté les moments de Thebichef dans leur cadre théorique, nous présentons un autre type de moment qui s'intègre dans le cadre des moments discrets, c'est le moment de Krawtchouk.

1.4.2 Les Moments de Krawtchouk

Les moments de Krawtchouk sont un ensemble formé à partir des polynômes de Krawtchouk, ceux-ci étaient introduits par *Mikhail Krawtchouk* [5], qui sont basés sur les distributions polynomiales. On peut contrôler la forme des polynômes de Krawtchouk avec la variation de la variable de probabilité p .

Dans [5], la définition des polynômes classiques de Krawtchouk de degré n est donnée par :

$$K_n(x; p, N) = \sum_{k=0}^N a_{k,n,p} x^k = {}_2F_1(-n, -x; -N; \frac{1}{p}) \quad (1.23)$$

avec $x, n = 0, 1, 2, \dots, N-1$, $N > 0$ et $p \in [0, 1]$. ${}_2F_1$ représente la fonction hypergéométrique donnée par :

$${}_2F_1(a, b; c; z) = \sum_{k=0}^{\infty} \frac{(a)_k (b)_k z^k}{(k!) (c)_k} \quad (1.24)$$

$(a)_k$ représente le symbole de Pochhammer donné par la formule(1.7). Étant donné que la fonction de poids est :

$$w(x, p, N-1) = \binom{N-1}{x} p^x (1-p)^{N-1-x}. \quad (1.25)$$

la condition d'orthogonalité est donnée par :

$$\sum_{x=0}^{N-1} w(x, p, N-1) K_n(x, p, N-1) K_m(x, p, N-1) = \rho(x; p, N-1) \delta_{nm}. \quad (1.26)$$

avec $n, m = 1, 2, \dots, N-1$, δ_{nm} c'est la fonction de Kronecher et le facteur de normalisation $\rho(x; p, N-1)$ présenté par :

$$\rho(x; p, N-1) = (-1)^x \left(\frac{1-p}{p} \right)^x \frac{n!}{(-N+1)_x}. \quad (1.27)$$

Afin d'éviter les erreurs de calcul prévues par la fonction hypergéométrique, en [6], ils ont présenté une relation récurrente pour les polynômes de Krawtchouk classiques, la formule est donnée par :

$$K_n(x, p, N-1) = \frac{(N-1)p - 2(n-1)p + n - 1}{p(N-n)} K_{n-1}(x, p, N-1) - \frac{1-p}{p} \frac{n-1}{N-n} K_{n-2}(x, p, N-1) \quad (1.28)$$

$$K_0(x, p, N-1) = 1 \quad (1.29)$$

$$K_1(x, p, N-1) = 1 - \frac{x}{(N-1)p} \quad (1.30)$$

Après la définition des polynômes de Krawtchouk, nous introduisons la formule de calcul de moment de Krawtchouk, pour l'ordre(n+m) qui est exprimée en fonction de polynômes

de Krawtchouk et des valeurs d'intensité de pixel exprimé par $f(x, y)$, la formule est donnée comme suit :

$$A_{mn} = \frac{1}{\rho(n, p_1, N-1)\rho(m, p_2, N-1)} \times \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} K_n(x, p_1, N-1)K_m(y, p_2, N-1)f(x, y) \quad (1.31)$$

les paramètres N et M représentent la taille de l'image suivant les deux axes X et Y .

Afin de rectifier le problème des erreurs de calcul au niveau des moments basés sur les polynômes classiques donnés par (1.28), dans [4] ils ont présenté les polynômes de Krawtchouk normalisés. ces derniers sont obtenus à partir de facteur de normalisation donné par l'équation(1.27), les polynôme normalisés sont donné par :

$$\tilde{K}_n(x, p, N-1) = \frac{K_n(x, p, N-1)}{\sqrt{\rho(n, p, N-1)}} \quad (1.32)$$

Pour encore diminuer les erreurs de calcul, les auteurs de [4] ils ont présenté une nouvelle forme des polynômes de Krawtchouk pondérés-normalisés donnée par :

$$\bar{K}_n(x, p, N-1) = K_n(x, p, N-1) \sqrt{\frac{w(x, p, N-1)}{\rho(n, p, N-1)}} \quad (1.33)$$

Avec cette modification la condition d'orthogonalité sera modifiée comme suit :

$$\sum_{x=0}^{N-1} \bar{K}_n(x, p, N-1)\bar{K}_m(x, p, N-1) = \delta_{nm}. \quad (1.34)$$

Remarque :

Il faut noter que les valeurs des polynômes de Krawtchouk pondérés-normalisés sont orthogonaux sur l'intervalle $[-1,1]$, cette propriété sera justifiée au niveau de la figure (1.3 (b)). Pour analyser l'influence de valeur de probabilité p sur différents ordres $n = 0, 1, 2$, on prend $p = 0.5$, afin de faire une variation de p avec Δp , si $p = 0.5 + \Delta p$, le polynômes de Krawtchouk sera décalé avec une approximation de $N\Delta p$, la direction de décalage dépend de signe de Δp , le décalage des polynômes de Krawtchouk pondérés-normalisés se fait en direction $+x$ si Δp positive et vice-versa, figure(1.4). L'importance de cette propriété sera utilisée pour l'extraction des caractéristiques importantes d'une image situées dans une région donnée.

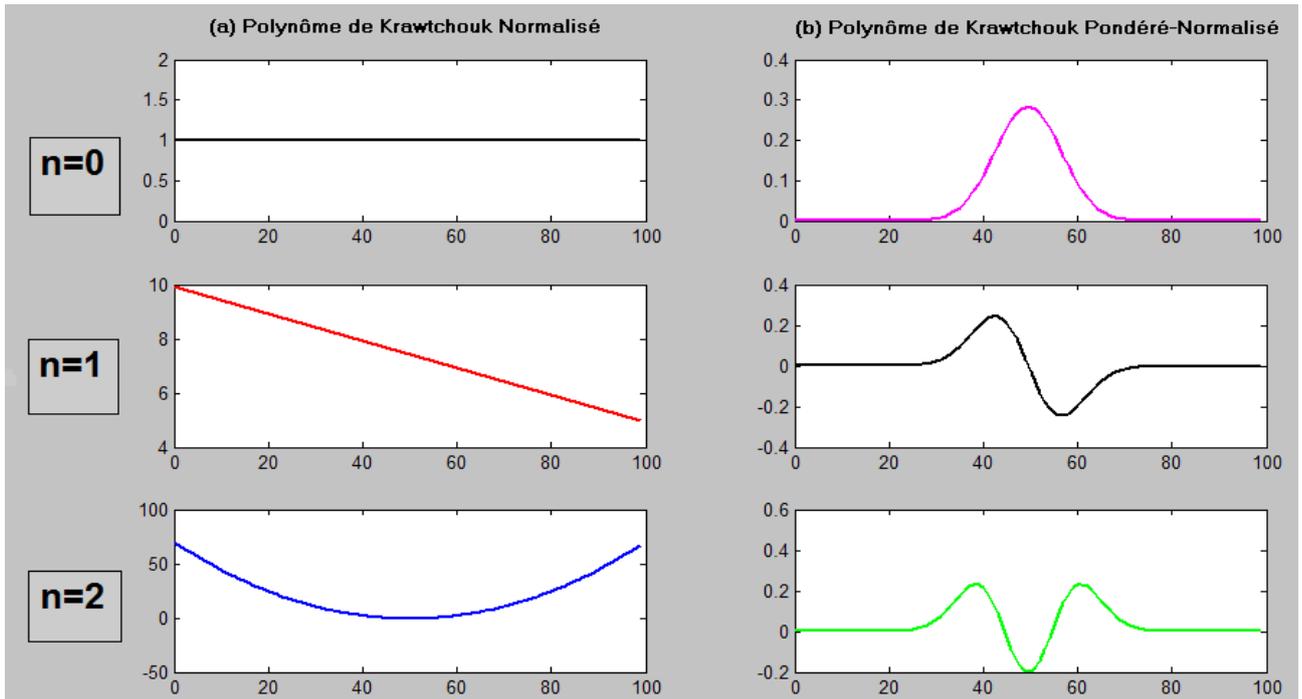


FIGURE 1.3 – le tracé de polynôme de Krawtchouk normalisé et pondéré-normalisé pour $p = 0.5$ et $N = 100$.

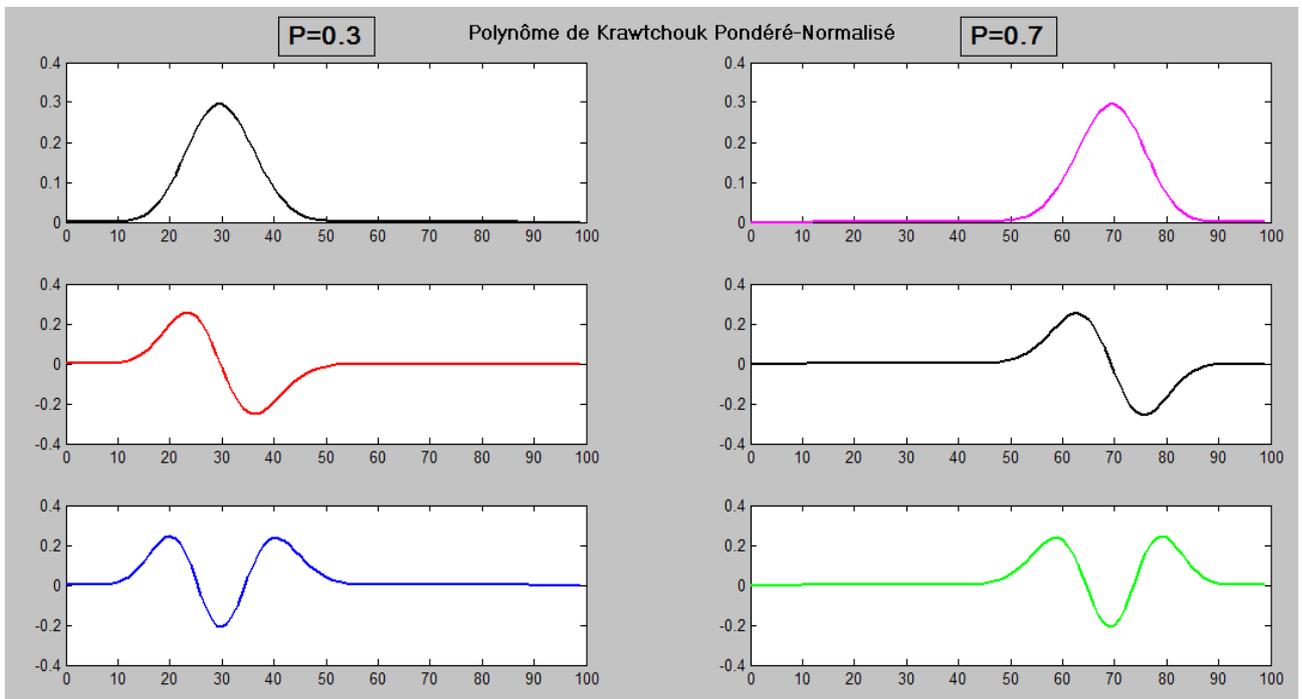


FIGURE 1.4 – Comparaison entre les polynômes de Krawtchouk pondérés-normalisés pour $p = 0.3$ et $p = 0.7$.

Dans l'objectif d'améliorer le calcul des polynômes de Krawtchouk pondérés-normalisés, dans [4] ils ont introduit une relation récurrente donnée par :

$$\bar{K}_n(x, p, N - 1) = A_n \bar{K}_{n-1}(x, p, N - 1) - B_n \bar{K}_{n-2}(x, p, N - 1) \quad (1.35)$$

$$\text{avec } A_n = \frac{(N - 1)p - 2(n - 1)p + n - 1 - x}{\sqrt{p(1 - p)n(N - n)}} \text{ et } B_n = \sqrt{\frac{(n - 1)(N - n + 1)}{(N - n)n}}$$

$$\bar{K}_0(x, p, N - 1) = \sqrt{w(x, p, N - 1)} \quad (1.36)$$

$$\bar{K}_1(x, p, N - 1) = \sqrt{w(x, p, N - 1)} \frac{(N - 1)p - x}{\sqrt{(N - 1)p(1 - p)}} \quad (1.37)$$

Après avoir donné les polynôme de Krawtchouk pondérés-normalisés, la formule de calcul de moment pour une image de taille $N \times N$ et d'intensité de pixel $f(x, y)$ sera modifiée comme suit :

$$Q_{mn} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \bar{K}_n(x, p_1, N - 1) \bar{K}_m(y, p_2, N - 1) f(x, y) \quad (1.38)$$

Comme les moments de Tchebichef, les moments de Krawtchouk s'intègre dans la partie de reconstruction des images, cette dernière se fait à partir des moments calculés de l'image originale, la forme de reconstruction est donné comme suit :

$$f(x, y) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \bar{K}_n(x, p_1, M - 1) \bar{K}_m(y, p_2, N - 1) Q_{nm}. \quad (1.39)$$

Les moments de Krawtchouk possèdent la propriété de symétrie, or cette propriété n'est validée que dans le cas de $p_1 = p_2 = 0.5$, la relation de symétrie basée sur les polynômes de Krawtchouk normalisés est donnée par :

$$\bar{K}_n(x, p, N - 1) = (-1)^n \bar{K}_n(N - 1 - x, p, N - 1) \quad (1.40)$$

Afin de mieux illustrer la propriété de symétrie, la figure (1.5) représente le tracé de cinq premiers ordres de polynôme de Krawtchouk Normalisé pour $N = 100$, cette propriété est appliquée aussi au niveau de calcul des moments de Krawtchouk, la formule est identique à (1.21) avec un changement de polynôme de Tchebichef avec le polynôme de Krawtchouk pondéré-normalisé.

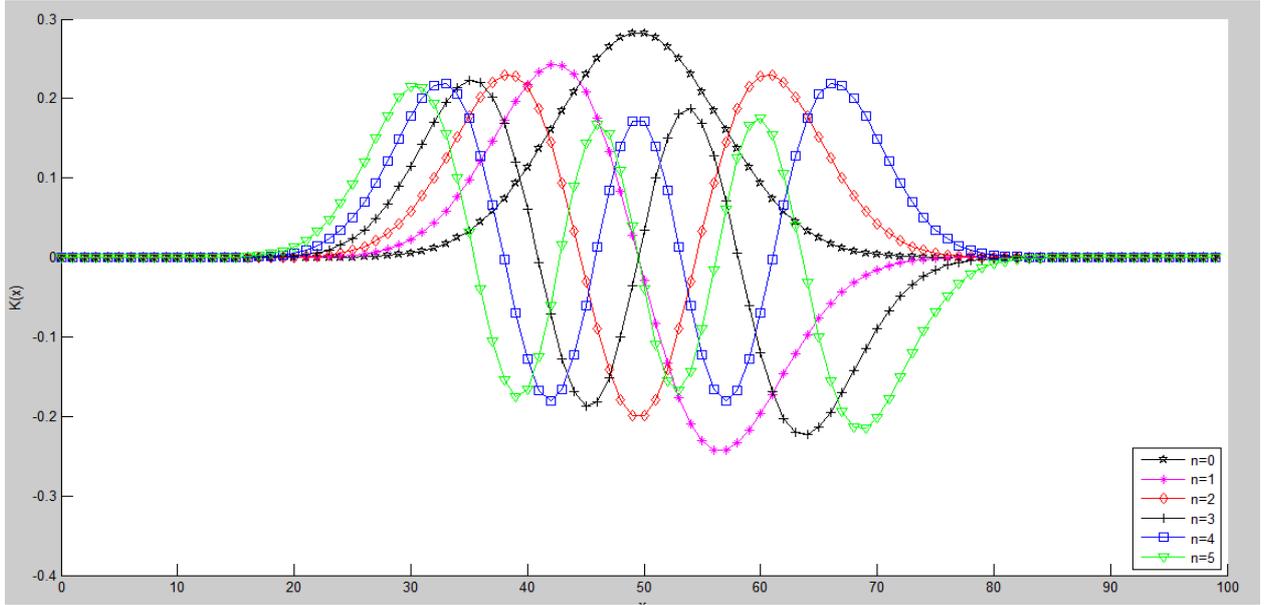


FIGURE 1.5 – le tracé de polynôme de Krawtchouk pondéré-normalisé pour $N=100$ et $p=0.5$.

Après avoir défini les différentes propriétés des polynômes et moments de Krawtchouk, nous continuons l'étude avec les moments discrets orthogonaux par la présentation de moment de Hahn.

1.4.3 Le Moment de Hahn

Les polynômes de Hahn appartiennent à l'ensemble des polynômes discrets et orthogonaux, ils sont similaires au polynôme de Tchebichef et Krawtchouk, mais ils sont assez complexes au niveau de calcul[7]. Ces polynômes sont associés à une distribution hypergéométrique négative, de plus ce type admet deux paramètres d'ajustement qui permettent le contrôle de forme soit au niveau de compression ou au niveau de reconstruction.

soit un entier $n \in [0, N]$ avec N est un entier positive, le polynôme classique de Hahn pour l'ordre n , $n=0,1,\dots,N-1$, est présenté en [7] comme suit :

$$h_n^{(\mu,\nu)}(x, N) = (N + \nu - 1)_n (N - 1)_n \quad (1.41)$$

$$\times \sum_{k=0}^n (-1)^k \frac{(-n)_k (-x)_k (2N + \mu + \nu - n - 1)_k}{(N + \nu - 1)_k (N - 1)_k} \frac{1}{k!}, \quad \text{avec } \mu, \nu > -1$$

$(a)_k$ représente le symbole de Pochhammer défini en (1.7), le rôle principale des paramètres d'ajustement μ et ν est de contrôler la forme des polynômes. Ces polynômes satisfont la condition d'orthogonalité donnée par :

$$\sum_{x=0}^{N-1} w(x) h_m^{(\mu,\nu)}(x, N) h_n^{(\mu,\nu)}(x, N) = d_n^2 \delta_{mn} \quad 0 \leq m, n \leq N - 1 \quad (1.42)$$

avec δ_{mn} représente la fonction de Kronecher, $w(x)$ c'est la fonction de pondération donnée par :

$$w(x) = \frac{1}{\Gamma(x+1)\Gamma(x+\mu+1)\Gamma(N+\nu-x)\Gamma(N-n-x)} \quad (1.43)$$

et la norme quadratique d_n^2 est donnée par :

$$d_n^2 = \frac{\Gamma(2N+\mu+\nu-n)}{\Gamma(2N+\mu+\nu-2n-1)\Gamma(N+\mu+\nu-n)} \times \frac{1}{\Gamma(N+\mu-n)\Gamma(N+\nu-n)\Gamma(n+1)\Gamma(N-n)} \quad (1.44)$$

Afin de remédier au problème des erreurs de calcul au niveau des moments causé par la définition des polynômes classiques de Hahn, Zhou et al en [7] ont présenté une nouvelle forme de polynôme de Hahn normalisé, donnée par :

$$\bar{h}_n^{(\mu,\nu)}(x, N) = h_n^{(\mu,\nu)}(x, N) \sqrt{\frac{w(x)}{d_n^2}}, \quad n = 0, 1, \dots, N-1 \quad (1.45)$$

Or avec cette modification au niveau des polynômes de Hahn, la condition d'orthogonalité sera décrite comme suit :

$$\sum_{x=0}^{N-1} \bar{h}_m^{(\mu,\nu)}(x, N) \bar{h}_n^{(\mu,\nu)}(x, N) = \delta_{mn} \quad 0 \leq m, n \leq N-1 \quad (1.46)$$

Remarque :

Les polynômes de Hahn normalisés admettent la propriété de symétrie, ce qui va influencer sur la complexité de calcul des polynômes normalisé de Hahn, la figure (1.6) illustre cette remarque, en plus les polynômes de Hahn sont orthogonaux dans l'intervalle $[-1,1]$.

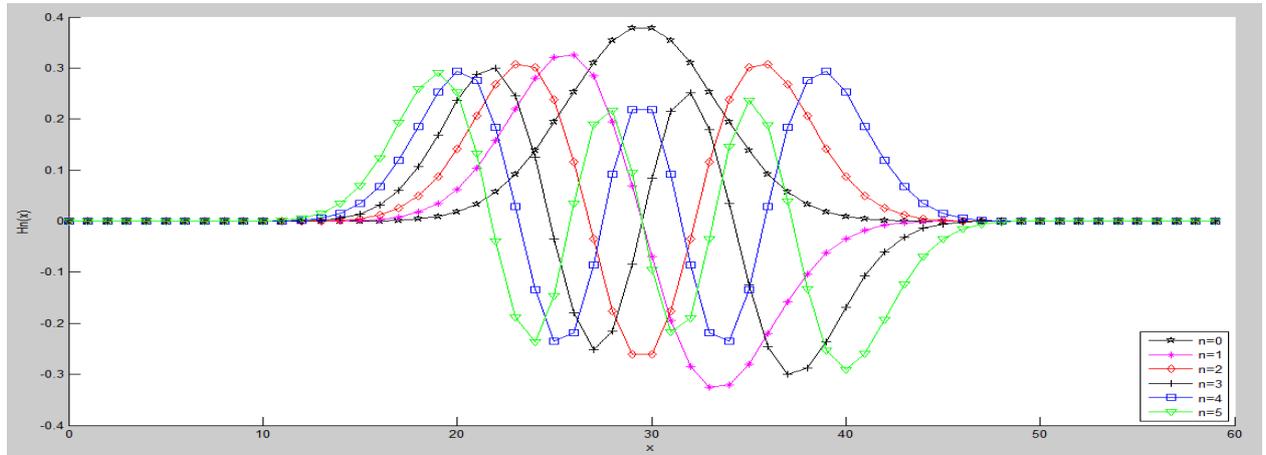


FIGURE 1.6 – le tracé de polynôme de Hahn normalisé pour $N=60$, $\nu = 0$ et $\mu = 0$.

Après l'introduction des polynômes normalisés de Hahn, Le moment de Hahn d'ordre $(n+m)$ pour une image de taille $N \times N$ est donné par :

$$H_{nm} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \bar{h}_m^{(\mu,\nu)}(x, N) \bar{h}_n^{(\mu,\nu)}(x, N) f(x, y) \quad \text{avec } n, m = 0, 1, \dots, N-1 \quad (1.47)$$

Comme nous l'avons introduit, les moments discrets offrent la possibilité de reconstruction des images à partir des moments calculés, comme les moments de Hahn appartiennent a ce type, la formule inverse de reconstruction est donnée par :

$$f(x, y) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} \bar{h}_m^{(\mu,\nu)}(x, N) \bar{h}_n^{(\mu,\nu)}(x, N) H_{mn} \quad (1.48)$$

Pour améliorer le calcul des polynômes de Hahn normalisés, les auteurs en [7] ont présenté une formule récursive de calcul des polynômes normalisés de Hahn :

$$\bar{h}_0^{(\mu\nu)}(x, N) = \sqrt{\frac{w(x)}{d_0^2}} \quad (1.49)$$

$$\bar{h}_1^{(\mu\nu)}(x, N) = \{(N + \nu - 1)(N - 1) - (2N + \mu + \nu - 2)x\} \sqrt{\frac{w(x)}{d_1^2}} \quad (1.50)$$

$$A\bar{h}_n^{(\mu\nu)}(x, N) = B\sqrt{\frac{d_{n-1}^2}{d_n^2}}\bar{h}_{n-1}^{(\mu\nu)}(x, N) + C\sqrt{\frac{d_{n-2}^2}{d_n^2}}\bar{h}_{n-2}^{(\mu\nu)}(x, N), \quad n = 2, 3, \dots, N-1 \quad (1.51)$$

avec :

$$A = -\frac{n(2N + \mu + \nu - n)}{(2N + \mu + \nu - 2n + 1)(2N + \mu + \nu - 2n)},$$

$$B = x - \frac{2(N-1) - \mu + \nu}{4} - \frac{(\mu^2 - \nu^2)(2N + \mu + \nu)}{4(2N + \mu + \nu - 2n + 2)(2N + \mu + \nu - 2n)},$$

$$C = \frac{(N-n+1)(N-n+\mu+1)(N-n+\nu+1)(N-n+\mu+\nu+1)}{(2N + \mu + \nu - 2n + 2)(2N + \mu + \nu - 2n + 1)}$$

Les polynômes discrets assurent la propriété de symétrie, la figure (1.6) représente cette propriété d'une façon expérimentale. Dans le même sens, la relation de symétrie est similaire à la formule (1.21) donnée au niveau des moments de Tchebichef, il suffit de changer les polynômes de Tchebichef normalisés par les polynômes de Hahn normalisés, donc pour une image de taille $N \times N$ on peut calculer que les premiers $N/2 \times N/2$ polynômes ce qui diminue la complexité de calcul .

Après avoir présenté les moments de Hahn dans leur cadre théorique, nous finalisons cette partie par l'introduction des moments de DualHahn.

1.4.4 Le Moment de DualHahn

Dans cette partie nous nous adressons à l'étude des polynômes discrets de DualHahn, ce type est orthogonal dans les non uniformes lattices (le lattice quadratique $X(s) = s(s+1)$), ce qui implique que ce nouvel ensemble de polynôme est le plus généralisé, car les polynômes de Tchebichef et Krawtchouk sont des cas spéciaux de DualHahn [8]. A cause de leur définition non uniforme lattice, les polynômes de DualHahn contenant plusieurs paramètres ce qui implique une variation d'erreur élevée au niveau de calcul.

Avant de présenter les polynômes de DualHahn, on fait un petit rappel concernant les polynômes de variable discret en non uniforme lattice. en [8], ils ont énoncé que l'ensemble des polynômes orthogonaux classiques sont modélisables comme des équations différentielles données par :

$$\tilde{\sigma}(x)y''(x) + \tilde{\tau}y'(x) + \lambda y(x) = 0 \quad (1.52)$$

Or cette équation n'est validée que pour les variables continus, pour la remettre valide pour les polynômes à variable discret en non uniforme lattice, nous exploitons le lattice quadratique $X(s) = s(s+1)$, d'où l'équation différentielle est représenté comme suit :

$$\tilde{\sigma}(x(s))\frac{\Delta}{\Delta X(s - \frac{1}{2})} \left[\frac{\nabla y(s)}{\nabla X(s)} \right] + \frac{\tilde{\tau}(X(s))}{2} \left[\frac{\Delta y(s)}{\Delta X(s)} + \frac{\nabla y(s)}{\nabla X(s)} \right] + \lambda y(s) = 0 \quad (1.53)$$

avec $\Delta X(s) = X(s+1) - X(s)$ et $\nabla X(s) = X(s) - X(s-1)$, représentent respectivement gradient et la dérivée de la différence finie de lattice quadratique. $\tilde{\sigma}(x(s))$ et $\tilde{\tau}(X(s))$ sont des polynômes pour variable $X(s)$ et de degré 2 et 1 respectivement, λ est une constante, alors la relation donné par (1.53) est réécrite comme suit :

$$\sigma(s)\frac{\Delta}{\Delta X(s - \frac{1}{2})} \left[\frac{\nabla y(s)}{\nabla X(s)} \right] + \tau(s) \left[\frac{\Delta y(s)}{\Delta X(s)} \right] + \lambda y(s) = 0 \quad (1.54)$$

avec

$$\sigma(s) = \tilde{\sigma}(x(s)) - \frac{1}{2}\tilde{\tau}(X(s)) \left[\Delta X(s - \frac{1}{2}) \right]. \quad (1.55)$$

$$\tau(s) = \tilde{\tau}(X(s)) \quad (1.56)$$

La solution d'équation(1.54) est donnée par le polynôme $y_n(X(s)) \equiv P_n(s)$ avec $P_n(s)$ dénoté par :

$$P_n(s) = \frac{B_n}{\rho(s)} \nabla_n^{(n)} [\rho_n(s)], \quad \nabla_n^{(n)} [\rho_n(s)] = \frac{\nabla}{\nabla X_1(s)} \cdots \frac{\nabla}{\nabla X_{n-1}(s)} \frac{\nabla}{\nabla X_n(s)} [\rho_n(s)] \quad (1.57)$$

$$(1.58)$$

avec $X_n(s) = X(s + \frac{n}{2})$, $\rho_n(s) = \rho_n(n+s) \prod_{k=1}^n \sigma(s+k)$, La solution $P_n(s)$ donnée par (1.57) satisfait la condition d'orthogonalité suivante :

$$\sum_{s=0}^{b-1} P_n(s)P_m(s)\rho(s) \left[\Delta X(s - \frac{1}{2}) \right] = \delta_m d_n^2. \quad (1.59)$$

d_n^2 exprime la norme quadratique pour les polynômes orthogonaux, et $\rho(s)$ représente la fonction non négative i.e -fonction

de

poinds- avec :

$$\rho(s) \left[\Delta X(s - \frac{1}{2}) \right] > 0, \quad a \leq s \leq b \quad (1.60)$$

Dans la suite nous présentons les polynômes de DualHahn comme un cas particulier des polynômes définis en non uniforme lattice. Les polynômes classiques de DualHahn notés par $w_n^{(c)}(s, a, b)$, $n = 0, 1, \dots, N - 1$, sont des solutions pour l'équation différentielle donnée en (1.54) avec :

$$\begin{aligned} \sigma(s) &= (s - a)(s + b)(s - c) \\ \tau(s) &= ab - ac + bc - a + b - c - 1 - X(s) \\ \lambda &= n \end{aligned} \quad (1.61)$$

et la fonction de poids est donnée par :

$$\rho(s) = \frac{\gamma(a + s + 1)\gamma(c + s + 1)}{\gamma(s - a + 1)\gamma(b - s)\gamma(b + s + 1)\gamma(s - c + 1)} \quad (1.62)$$

avec $-1/2 < a < b$, $|c| < a + 1$ et $b = a + N$.

Remarque :

Si on admet $X(s) = s$, et les paramètres a, b et c définis par $a = (\alpha + \beta)/2$, $b = a + N$, $c = (\beta - \alpha)/2$, dans ce cas les polynômes de DualHahn seront équivalents aux polynômes de Hahn $h_n^{\alpha, \beta}(x, N)$. Si on met $\alpha = 0$ et $\beta = 0$, les polynômes de Hahn seront réduits à ceux de Tchebichef, de plus si on admet que $\beta = pt$ et $\alpha = (1 - p)t$, pour $t \rightarrow \infty$ on obtient les polynômes de Krawtchouk $K_n(x; p, N)$.

le polynôme de DualHahn d'ordre n est donné par la formule suivante :

$$w_n^{(c)}(s, a, b) = \frac{(a - b + 1)_n (a + c + 1)_n}{n!} {}_3F_2(-n, a - s, a + s + 1; a - b + 1, a + c + 1; 1) \quad (1.63)$$

$$n = 0, 1, \dots, N - 1, \quad s = a, a + 1, \dots, b - 1.$$

$(u)_k$ définit le symbole de Pochhammer (1.7) et ${}_3F_2(a_1, a_2, a_3; b_1, b_2, z)$ représente le fonction hypergéométrique (1.6). Les polynômes de DualHahn satisfont la propriété d'orthogonalité donnée par :

$$\sum_{s=0}^{b-1} w_n^{(c)}(s, a, b)(s) w_m^{(c)}(s, a, b) \rho(s) \left[\Delta X(s - \frac{1}{2}) \right] = \delta_{nm} d_n^2. \quad (1.64)$$

où $\rho(s)$ exprime la fonction de poids donnée en (1.62) et

$$d_n^2 = \frac{\gamma(a + c + n + 1)}{n!(b - a - n - 1)!\gamma(b - c - n)} \quad n = 0, 1, \dots, N - 1 \quad (1.65)$$

Avant d'introduire la relation de calcul pour générer l'ensemble de polynômes de DualHahn, nous présentons les polynômes de DualHahn normalisés, dont l'objectif est de remédier à la divergence des valeurs au niveau de calcul de polynômes classiques. Les auteurs en [8] ont présenté une formule normalisée donnée par :

$$\hat{w}_n^c(s, a, b) = w_n^c(s, a, b) \sqrt{\frac{\rho(s)}{d_n^2} \left[\Delta X(s - \frac{1}{2}) \right]} \quad n = 0, 1, \dots, N - 1 \quad (1.66)$$

avec cette modification la condition d'orthogonalité revient comme suit :

$$\sum_{s=0}^{b-1} \hat{w}_n^{(c)}(s, a, b) \hat{w}_m^{(c)}(s, a, b) = \delta_{nm}, \quad \text{avec } n, m = 0, 1, \dots, N - 1. \quad (1.67)$$

Les valeurs de polynômes normalisés sont limitées dans l'intervalle $[-1,1]$, la figure (1.7) représente le tracé de cinq premiers ordres pour $N = b = 40$ et $a = c = 0$.

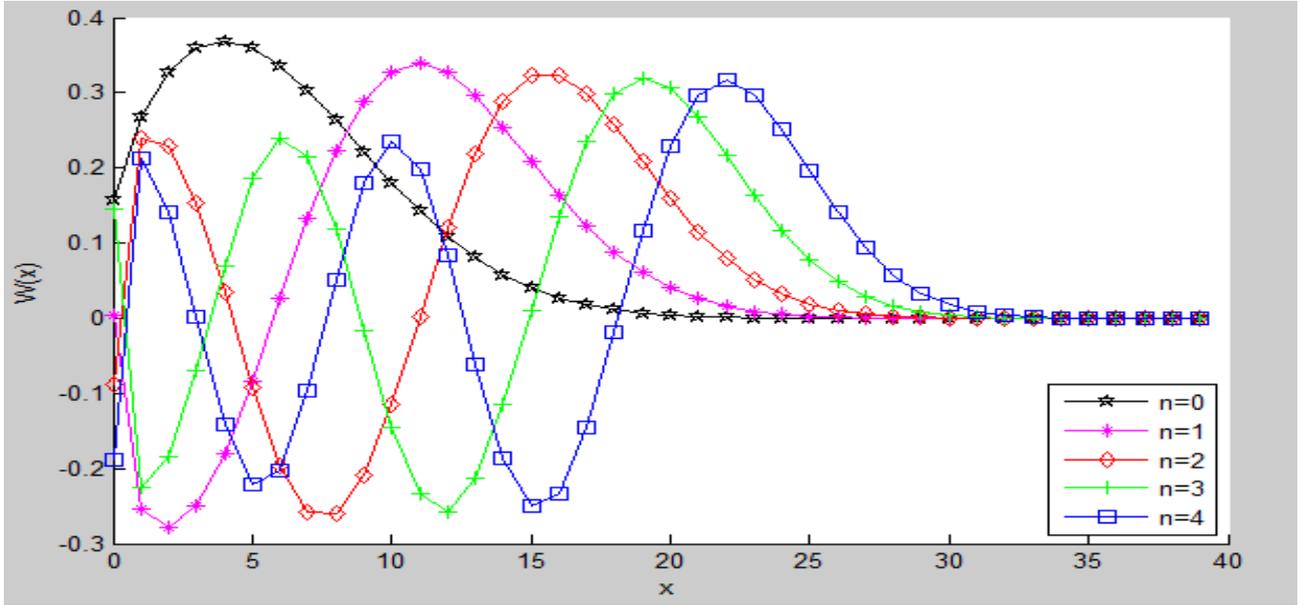


FIGURE 1.7 – le tracé de polynôme de DualHahn normalisé pour $N=b=40$, $a=c=0$.

Les moments de dual Hahn sont formés à partir des polynômes normalisés de DualHahn. Considérons une image de dimension $N \times N$ et d'intensité de pixel $f(s, t)$, l'ordre du moment $(n + m)$ de dual Hahn est donné par :

$$W_{nm} = \sum_{s=a}^{b-1} \sum_{t=a}^{b-1} \hat{w}_n^{(c)}(s, a, b) \hat{w}_m^{(c)}(t, a, b) f(s, t) \quad (1.68)$$

La propriété d'orthogonalité de DualHahn, permet de garantir le pouvoir de reconstruction, l'image reconstruite est obtenue à partir de formule inverse définie par :

$$f(s, t) = \sum_{n=0}^{N-1} \sum_{m=0}^{b-1} \hat{w}_n^{(c)}(s, a, b) \hat{w}_m^{(c)}(t, a, b) W_{nm} \quad (1.69)$$

Après avoir présenté les moments de DualHahn, les auteurs en [8] ont présenté une relation récursive pour calculer les polynômes de DualHahn, cet algorithme est introduit afin d'éviter les erreurs de calcul prévues par la fonction hypergéométrique et la fonction de Gamma, la relation récursive est donnée par :

$$\hat{w}_n^{(c)}(s, a, b) = A \frac{d_{n-1}}{d_n} \hat{w}_{n-1}^{(c)}(s, a, b) + B \frac{d_{n-2}}{d_n} \hat{w}_{n-2}^{(c)}(s, a, b) \quad (1.70)$$

où

$$A = \frac{1}{n} [s(s+1) - ab + ac - bc - (b-a-c-1)(2n-1) + 2(n-1)^2]$$

$$B = -\frac{1}{n} (a+c+n-1)(b-a-n+1)(b-c-n+1)$$

$$\text{avec } \hat{w}_0^{(c)}(s, a, b) = \sqrt{\frac{\rho(s)}{d_0^2} \left[\Delta X(s - \frac{1}{2}) \right]} \quad (1.71)$$

$$\hat{w}_1^{(c)}(s, a, b) = \frac{1}{\rho(s)} \frac{\rho_1(s-1) - \rho_1(s)}{X(s+1/2) - X(s-1/2)} \sqrt{\frac{\rho(s)}{d_1^2} \left[\Delta X(s - \frac{1}{2}) \right]} \quad (1.72)$$

Comme les autres moments introduits en avant, Les polynômes de DualHahn garantissent la propriété de symétrie, ce qui va influencer sur le temps de calcul au niveau de polynômes, la propriété de symétrie est donnée par :

$$\hat{w}_n^{(c)}(s, a, b) = (-1)^s \hat{w}_{N-1-n}^{(c)}(s, a, b) \quad (1.73)$$

Remarque :

Nous cherchons d'abord l'influence de paramètre a au niveau de calcul des polynômes, nous prenons $N = 40$, $a = c$ et $b = N + a$, dans les différents graphes avec $a = 0$, $a = 5$ et $a = 12$, la figure (1.8) montre que les polynômes se déplacent de gauche à droite quand les valeurs de paramètre a s'augmentent.

Ensuite nous essayons de montrer l'influence de paramètre c au niveau de calcul des polynômes de DualHahn normalisés, pour cela en prend $N=40$ et on fait un choix arbitraire pour a , on lui affecte la valeur 8, et nous faisons cinq cas pour le paramètre c où $c = -8$, $c = -4$, $c = 0$, $c = 4$, $c = 8$, la figure (1.9) montre que les polynômes se déplacent de gauche à droite quand c augmente.

1.5 Conclusion

Dans ce chapitre, nous avons présenté les quatre types de moments discrets dans leur cadre théorique, en particulier, les méthodes de calcul des moments, les formules de reconstruction et les propriétés admises pour chaque type.

Pour ce qui suit, nous implémentons ces différents types de moments discrets, dont le but de faire une étude comparative entre les quatre moments en termes de pouvoir de reconstruction.

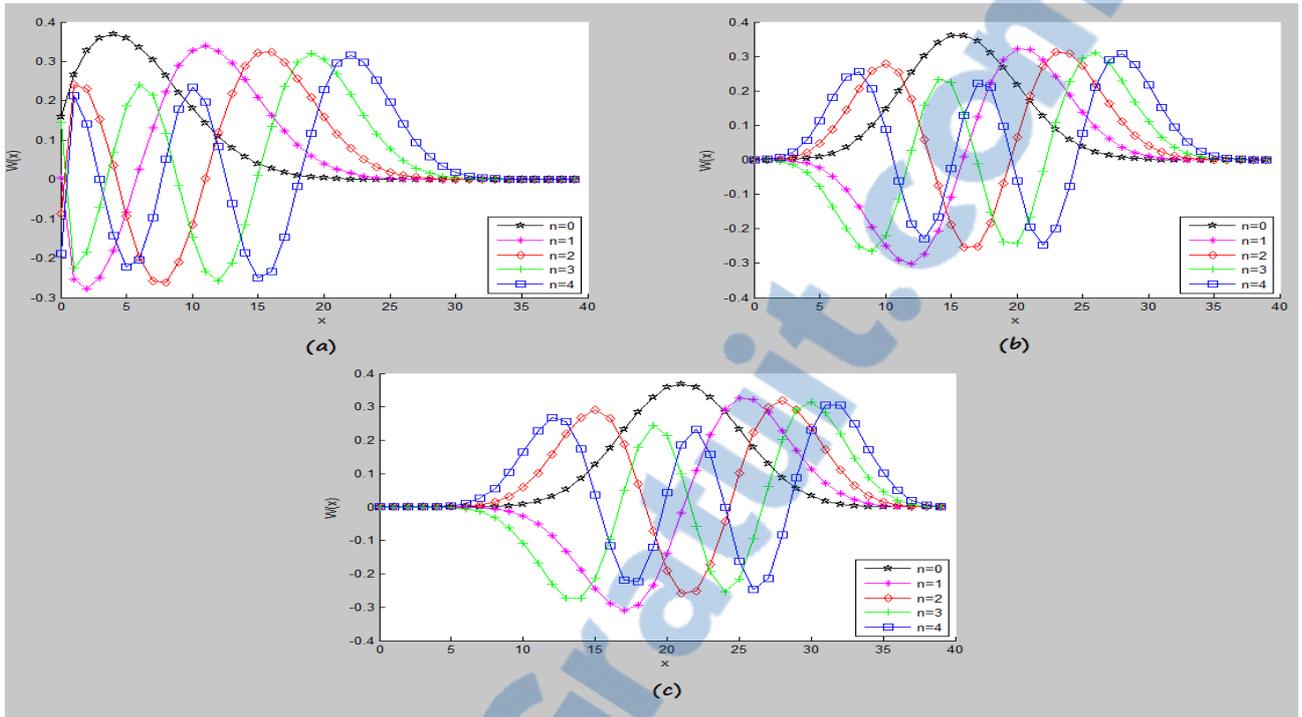


FIGURE 1.8 – l'influence de paramètre $-a-$ pour les polynômes normalisés de DualHahn avec $b = N+a$, $c=a$ et (a) $a = 0$, (b) $a=-5$, (c) $a = 12$

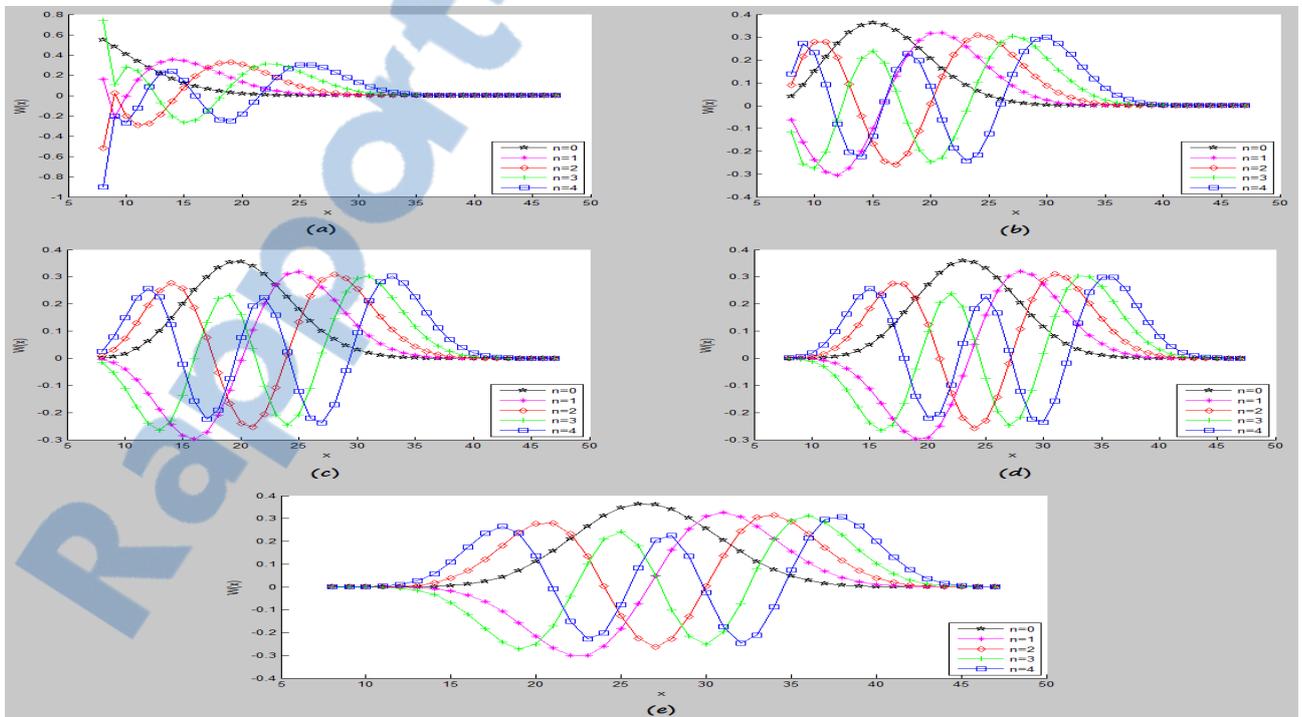


FIGURE 1.9 – l'influence de paramètre $-c-$ pour les polynômes normalisés de DualHahn, $a = 8$, $b = 48$. (a) $c = -8$, (b) $c = -4$, (c) $c = 0$, (d) $c = 4$, (e) $c = 8$

Chapitre 2

Reconstruction des Images 2-D à partir des Moments Discrets

2.1 Introduction

La reconstruction des images 2-D s'effectue en deux phases, la phase de transformation et la phase de reconstruction :

- **Phase de transformation :**

A ce niveau les polynômes orthogonaux discrets jouent le rôle de transformation des intensités de pixel d'une image à une image contenant des moments, qu'on l'appelle *image moment*.

- **Phase de reconstruction :**

En cette phase, la propriété d'orthogonalité des polynômes est utilisée au niveau de la transformation inverse, dont l'objectif est de transformer les valeurs d'image moment en intensité des pixels.

En effet, au niveau de la phase de décomposition, l'image des moments stocke l'information des distributions fréquentielles obtenues à partir des intensité des pixels. D'autre part, les caractéristiques globales d'une image sont obtenues à partir des moments de premier ordre, pour cette raison, les moments d'ordre supérieur n'influencent pas sur l'image reconstruite, car ils ne contiennent que le minimum d'informations [6].

Cette partie, prend en charge l'implémentation des différents algorithmes proposés par les moments discrets introduits au premier chapitre, et en même temps nous exprimons des résultats de comparaison, soit dans un cadre particulier ou général.

2.2 Étude Expérimentale

2.2.1 Méthode

Mesurer la qualité d'une image représente une problématique au niveau de traitement d'image, dans notre étude la mesure de qualité sera approximée par la mesure d'erreur de reconstruction par rapport à l'image origine. Pour cela nous admettons le PSNR (Peak Signal to Noise Ratio) comme outil de mesure de qualité de l'image reconstruite. Le PSNR c'est une mesure basée sur une fonction logarithmique qui a pour variable le MSE, qui définit l'erreur quadratique moyenne, que l'on définit par :

$$MSE = \frac{1}{N^2} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} (K_{x,y} - I_{x,y})^2. \quad (2.1)$$

Sachant que $I_{x,y}$ et $K_{x,y}$ représentent respectivement l'intensité de pixel pour l'image origine et reconstruite.

Si on admet que MAX_i représente la valeur maximale d'intensité de pixel d'une image (i.e 255 dans notre étude), le PSNR est défini par :

$$PSNR = 10 \log_{10} \left(\frac{MAX_i^2}{MSE} \right). \quad (2.2)$$

Comme le PSNR est calculé à partir de logarithme inverse du MSE, logiquement, Une valeur faible du MSE signifie qu'il y a moins de différence entre les deux images comparées, donc plus la valeur du PSNR est grande, les images sont presque identiques.

2.2.2 Reconstruction des images 2-D à partir de Moments Tchebichef

Comme nous avons déjà défini précédemment, les moments de Tchebichef offrent le pouvoir de reconstruction, avant de présenter les différentes expérimentations concernant la reconstruction des images 2-D en niveaux de gris ou binaires, nous présentons les algorithmes de calcul des polynômes de Tchebichef, des moments de Tchebichef et de reconstruction des images, qui sont illustrés comme suit :

Algorithm 1: Calcul des polynômes normalisés discrets de Tchebichef**Input:** N un entier positif exprime la taille de l'image suivant X et Y**Output:** T matrice de polynômes de Tchebichef normalisés

```
1 for  $i \leftarrow 0$  to  $N - 1$  do
2    $T_0(i) = 1$  ;
3    $T_1(i) = \frac{2i+1-N}{N}$  ;
4   for  $n \leftarrow 2$  to  $N - 1$  do
5      $T_n(i) = \frac{(2n-1)T_1(i)T_{n-1}(i) - (n-1)(1 - \frac{(n-1)^2}{N^2})T_{n-2}(i)}{n}$  ;
6 Return T
```

Algorithm 2: Calcul des moments discrets de Tchebichef**Input:** $Nmax < N$ un entier positif exprime l'ordre de reconstruction suivant l'axe X et Y, f l'image 2-D en niveau de gris de taille $N*N$, t matrice de polynôme**Output:** T matrice de moment

```
1 for  $n \leftarrow 0$  to  $Nmax$  do
2   for  $m \leftarrow 0$  to  $Nmax$  do
3      $sum \leftarrow 0$ 
4     for  $x \leftarrow 0$  to  $N - 1$  do
5       for  $y \leftarrow 0$  to  $N - 1$  do
6          $sum = sum + (t_n(x)t_m(y)f(x, y))$ ;
7      $T_{nm} = sum$ ;
8 Return t
```

Algorithm 3: Reconstruction des images 2-D en niveaux de gris**Input:** $Nmax < N$ un entier positif exprime l'ordre de reconstruction suivant l'axe X et Y, T matrice des moment, t matrice de polynômes**Output:** *image_reconstruite*

```
1 for  $x \leftarrow 0$  to  $N$  do
2   for  $y \leftarrow 0$  to  $N$  do
3      $sum \leftarrow 0$ 
4     for  $n \leftarrow 0$  to  $Nmax$  do
5       for  $m \leftarrow 0$  to  $Nmax$  do
6          $sum = sum + (t_n(x)t_m(y)T_{nm})$ ;
7      $image\_reconstruite(x, y) = sum$ ;
8 Return image_reconstruite
```

Après l'introduction des algorithmes de calcul, nous présentons l'étude expérimentale concernant la reconstruction des images 2-D binaires, et niveau de gris.

- **Image 2-D binaire :**

Dans cette étude nous prenons comme exemple, l'image de la lettre E et chiffre 7 pris de la base de données MNIST. Ces deux images sont de forme carrées de taille 20×20 .

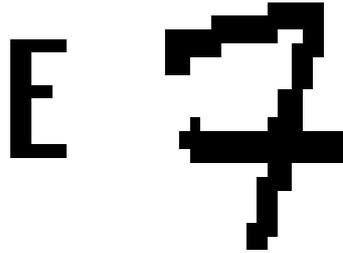


FIGURE 2.1 – image binaire de chiffre 7 et lettre E

La reconstruction des deux images pour différents ordres est donnée comme suit :

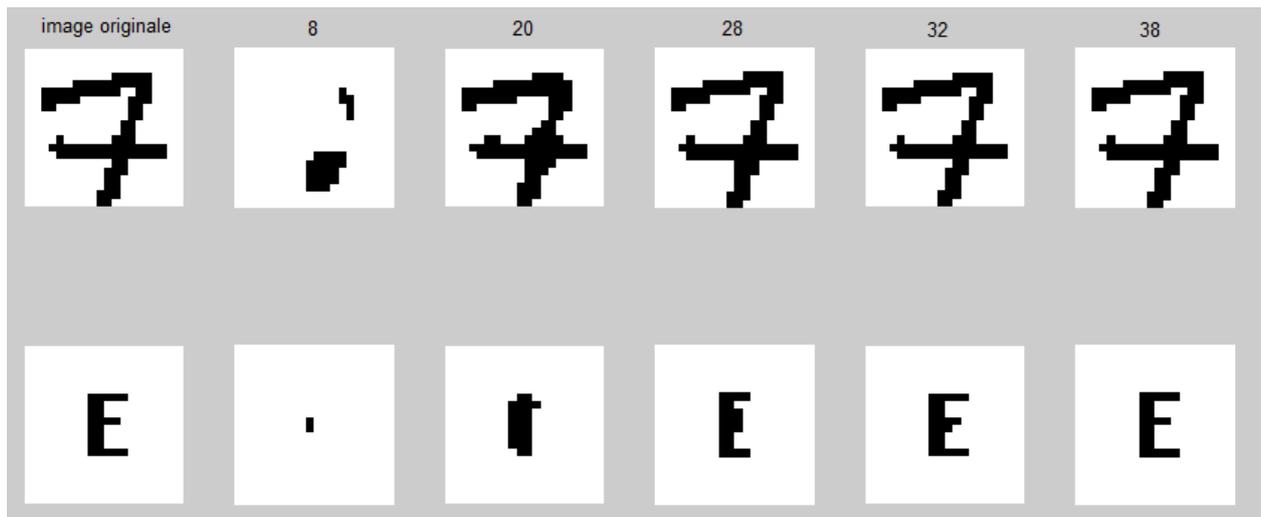


FIGURE 2.2 – reconstruction des images binaires pour l'ordre $n+m=8$, $n+m=20$, $n+m=28$, $n+m=32$ et $n+m=38$.

La sortie de reconstruction visualisée au-dessus, montre qu'à partir de l'ordre 20 l'information reconstruite a une signification par rapport à l'information originale. Cependant pour évaluer le taux d'erreur de reconstruction, les deux courbes présentées en figure (2.3 et 2.4), expriment l'erreur de reconstruction obtenue en différents ordres, de même on peut remarquer qu'à partir d'ordre 20, l'erreur de reconstruction se diminue.

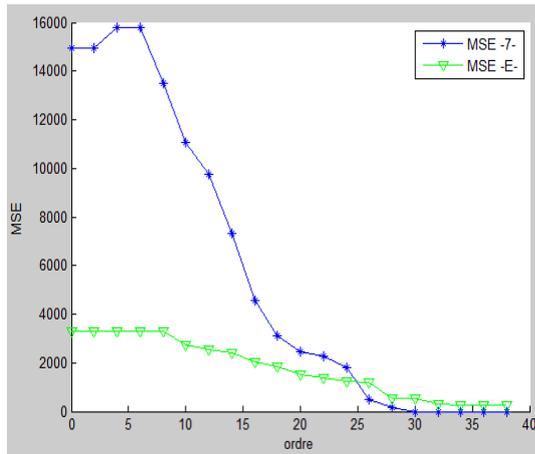


FIGURE 2.3 – la courbe de MSE pour différents ordres de reconstruction du chiffre -7- et la lettre -E-

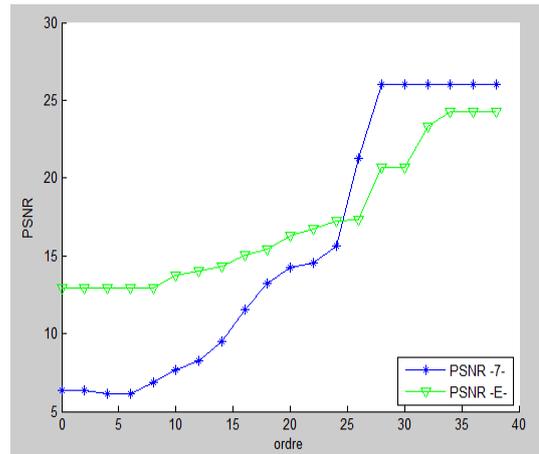


FIGURE 2.4 – la courbe de PSNR pour différents ordres de reconstruction du chiffre -7- et la lettre -E-

- **Image 2-D niveau de gris -cas sans bruit- :**

Dans cette partie nous évaluons la robustesse de reconstruction des images 2-D en niveau de gris via les moments discrets de Tchebichef, pour cela nous prenons comme image de test, trois images, Lena, Cameraman et papper de taille 64×64 , figure(2.5), les figures (2.6, 2.7 et 2.8) illustrent les différents ordres de reconstruction, le taux d'erreur sera exprimé par des courbes afin de trouver l'ordre optimum où l'information reconstruite aura une signification.



FIGURE 2.5 – image d'expérimentation, Cameramman, Lena, Papper, de taille 64×64 .



FIGURE 2.6 – Reconstruction des images niveau de gris pour différents ordres

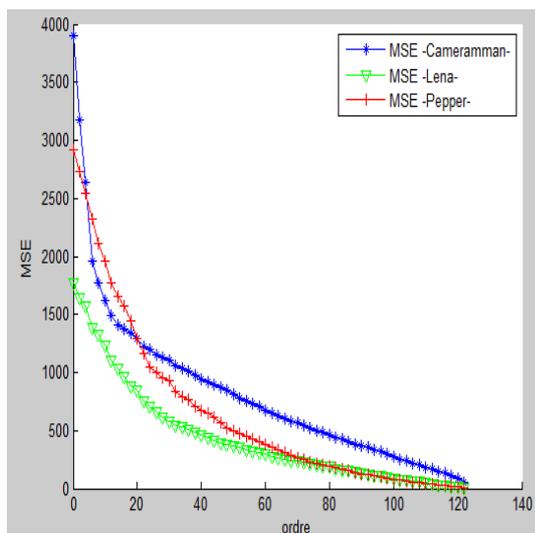


FIGURE 2.7 – la courbe de MSE pour différents ordres de reconstruction

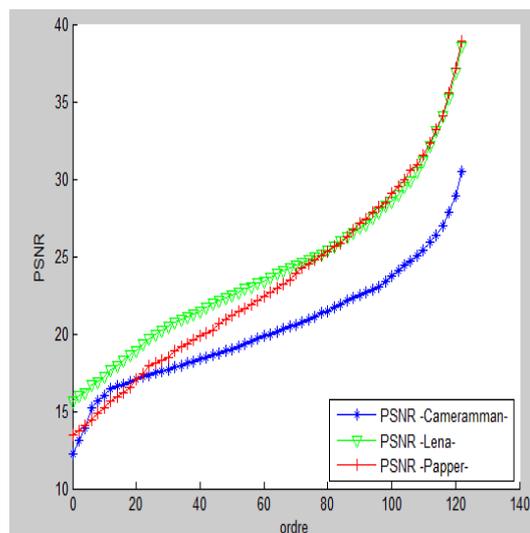


FIGURE 2.8 – la courbe de PSNR pour différents ordres de reconstruction

D'après les deux courbes MSE et PSNR on peut interpréter qu'au niveau des deux images Lena et Pepper, le taux d'erreur se diminue dès que l'ordre atteint la valeur 80, pour le Cameramman le taux d'erreur de reconstruction n'a une valeur significatif que dès l'ordre atteint les valeurs proche de $2N - 2$.

- **Image 2-D niveau de gris -cas avec bruit- :**

Pour évaluer le taux d'erreur de reconstruction au niveau des images 2-D niveau de gris avec bruit, nous suggérons une expérimentation qui présente la reconstruction et évalue le taux d'erreur à partir des courbes qui l'analysent. Dans notre étude nous présentons deux types de bruit, bruit impulsionnel et bruit gaussien.

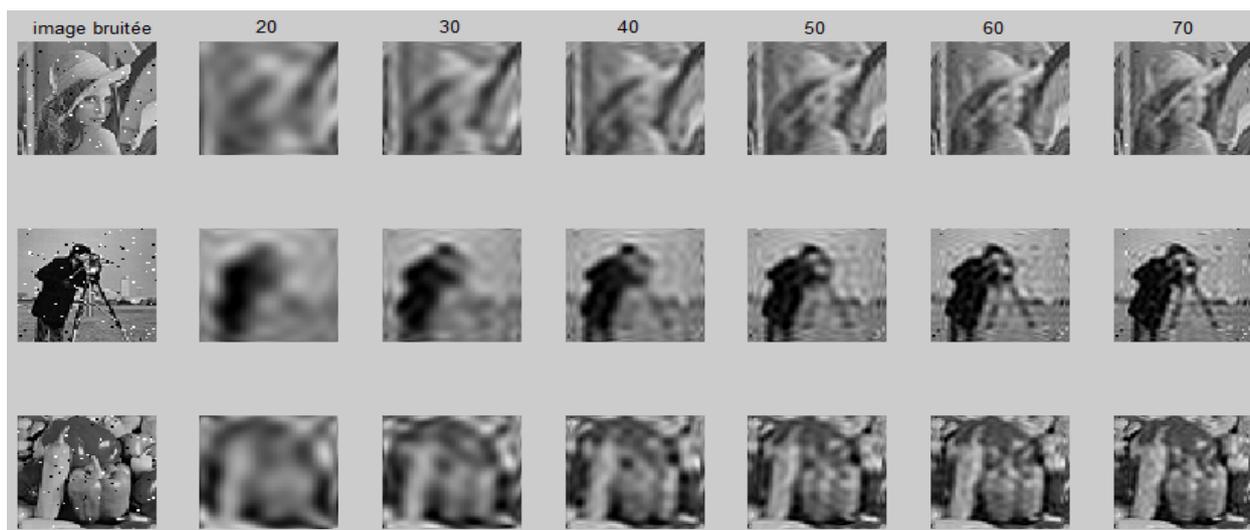


FIGURE 2.9 – la reconstruction des images bruitées avec bruit impulsionnel de densité $d=0.02$.

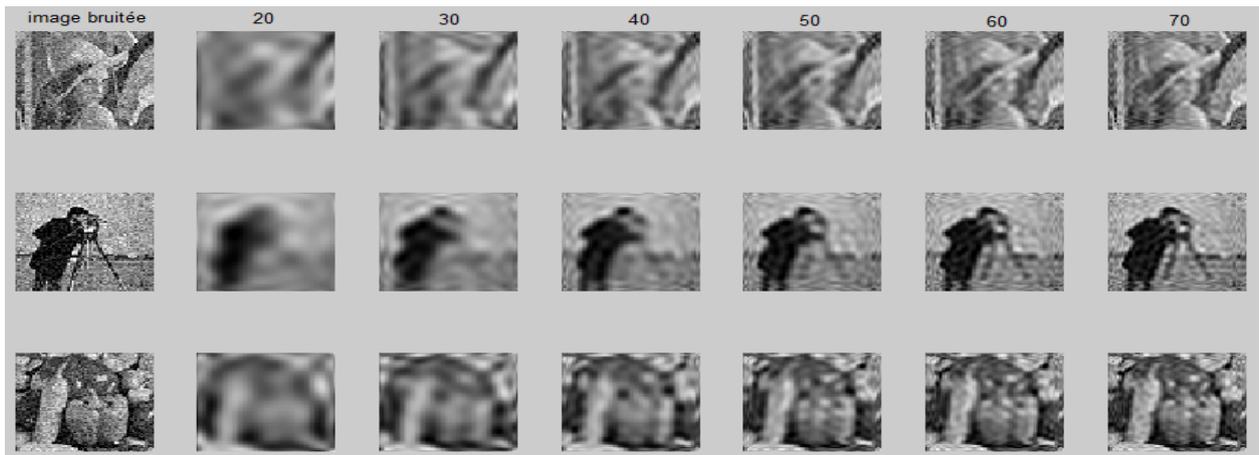


FIGURE 2.10 – la reconstruction des images bruitées avec bruit gaussienne de densité $d=0.02$.

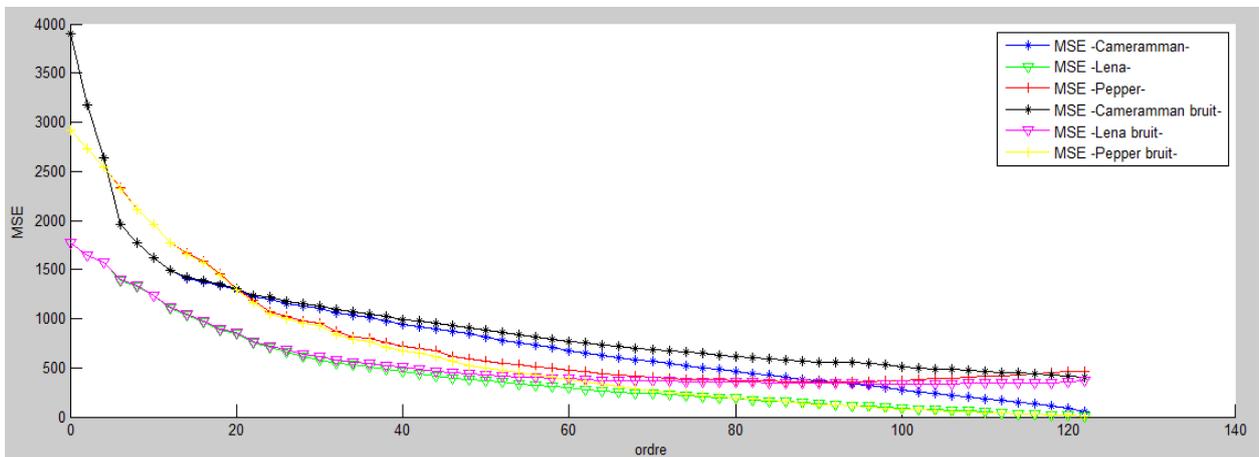


FIGURE 2.11 – le taux d'erreur de reconstruction des images bruitées avec bruit impulsional de densité $d=0.02$.

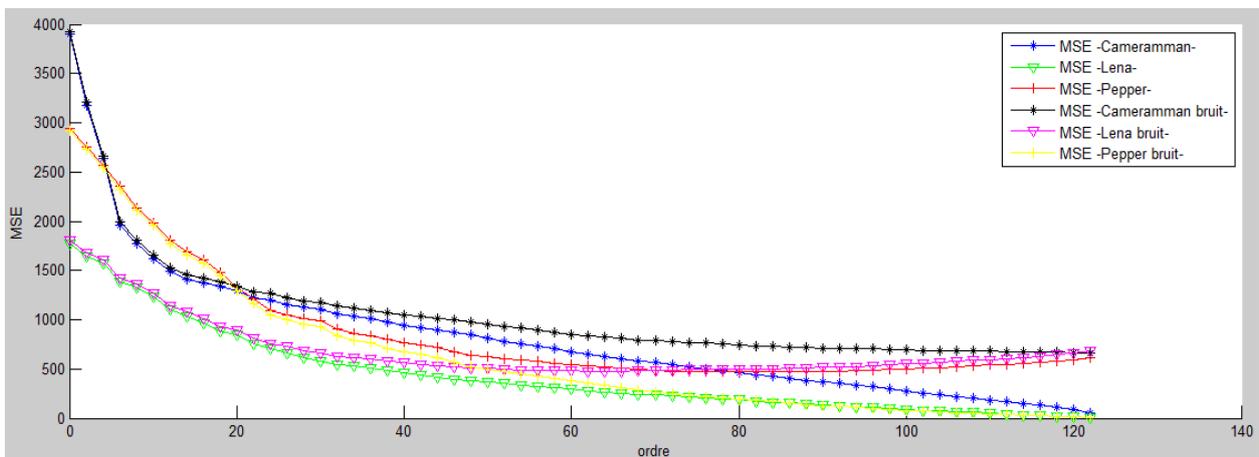


FIGURE 2.12 – le taux d'erreur de reconstruction des images bruitées avec bruit gaussien de densité $d=0.02$

Après la présentation des expérimentations concernant la reconstruction des images 2-D via les moments de Tchebichef, nous entamons une seconde partie, concernant les expérimentations de reconstruction des images 2-D à partir des moments de Krawtchouk.

2.2.3 Reconstruction des images 2-D à partir des Moments de Krawtchouk

En suivant la même démarche de la partie précédente, nous commençons par la présentation des algorithmes de calcul des polynômes de Krawtchouk, des moments de Krawtchouk et de reconstruction des images 2-D (binaire et niveau de gris) à partir des moments de Krawtchouk, les algorithmes de calcul sont donnés comme suit :

Algorithm 4: Calcul des polynômes discrets de Krawtchouk	
Input: N un entier positif exprime la taille de l'image, $p \in [0, 1]$ variable de probabilité, w la fonction de poids.	
Output: K matrice de polynômes de Krawtchouk normalisés	
1	for $i \leftarrow 0$ to $N - 1$ do
2	$K_0(i) = \sqrt{w(i; p, N - 1)}$;
3	$K_1(i) = \sqrt{w(i; p, N - 1)} \frac{(N-1)p-i}{\sqrt{(N-1)p(1-p)}}$;
4	for $n \leftarrow 2$ to $N - 1$ do
5	$A_n = \frac{(N-1)p-2(n-1)p+n+1}{\sqrt{p(1-p)n(N-n)}}$; $B_n = \sqrt{\frac{(n-1)(N-n+1)}{(N-n)n}}$;
6	$K_n(i) = A_n K_{n-1}(i) - B_n K_{n-2}(i)$;
7	Return K

Algorithm 5: Calcul des moments discrets de Krawtchouk	
Input: $Nmax < N$ un entier positif exprime l'ordre de reconstruction, f l'image 2-D en de taille $N \times N$, k matrice de polynôme	
Output: K matrice de moment	
1	for $n \leftarrow 0$ to $Nmax$ do
2	for $m \leftarrow 0$ to $Nmax$ do
3	$sum \leftarrow 0$
4	for $x \leftarrow 0$ to $N - 1$ do
5	for $y \leftarrow 0$ to $N - 1$ do
6	$sum = sum + (k_n(x)k_m(y)f(x, y))$;
7	$K_{nm} = sum$;
8	Return K

Algorithm 6: Reconstruction des images 2-D en niveaux de gris

Input: $Nmax < N$ un entier positif exprime l'ordre de reconstruction , K la matrice des moment, k la matrice de polynômes

Output: $image_reconstruite$

```

1 for  $x \leftarrow 0$  to  $N$  do
2   for  $y \leftarrow 0$  to  $N$  do
3      $sum \leftarrow 0$ 
4     for  $n \leftarrow 0$  to  $Nmax$  do
5       for  $m \leftarrow 0$  to  $Nmax$  do
6          $sum = sum + (k_n(x)k_m(y)K_{nm});$ 
7        $image\_reconstruite(x, y) = sum;$ 
8 Return  $image\_reconstruite$ 

```

Après la présentation des différents algorithmes de calcul, nous présentons les différentes expérimentations concernant la reconstruction des images 2-D via les moments de Krawtchouk, mais au début nous analysons l'influence de paramètre p sur la reconstruction des images.

- **Extraction des caractéristiques des régions importantes**

Dans cette partie nous discutons l'importance des variable p_1 et p_2 dans la variation d'extraction des caractéristiques d'une image, le paramètre de probabilité p_1 utilisé pour l'extraction des caractéristiques horizontales, si $p_1 < 0.5$ l'extraction des caractéristiques se fait de gauche vers la droite, si $p_1 > 0.5$ l'extraction des caractéristiques se fait inversement, le paramètre de probabilité p_2 sera utilisé pour l'extraction des caractéristiques verticales, si $p_2 < 0.5$ l'extraction des caractéristique se fait de haut vers le bas, si $p_2 > 0.5$ l'extraction des caractéristique se fait inversement. La figure (2.13) illustre l'extraction des caractéristiques d'une image contenant quatre lettres :

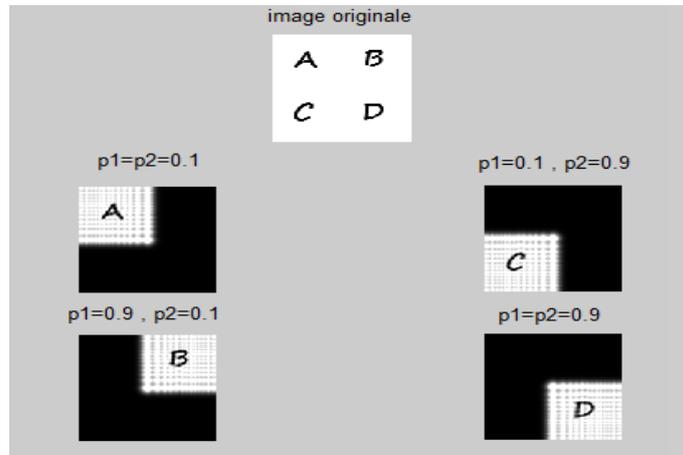


FIGURE 2.13 – Extraction des caractéristiques pour l'ordre $n+m=60$.

Dans la suite des expérimentations nous fixons les deux paramètres de probabilité p_1 et p_2 à 0.5, afin de garantir l'intérêt de symétrie au niveau de calcul des polynômes et moment.

- **Image 2-D binaire :**

Dans cette partie, nous présentons les différents ordres de reconstruction basés sur les moments de Krawtchouk, les images appliquées pour cette partie sont identiques aux images données par la figure (2.1).

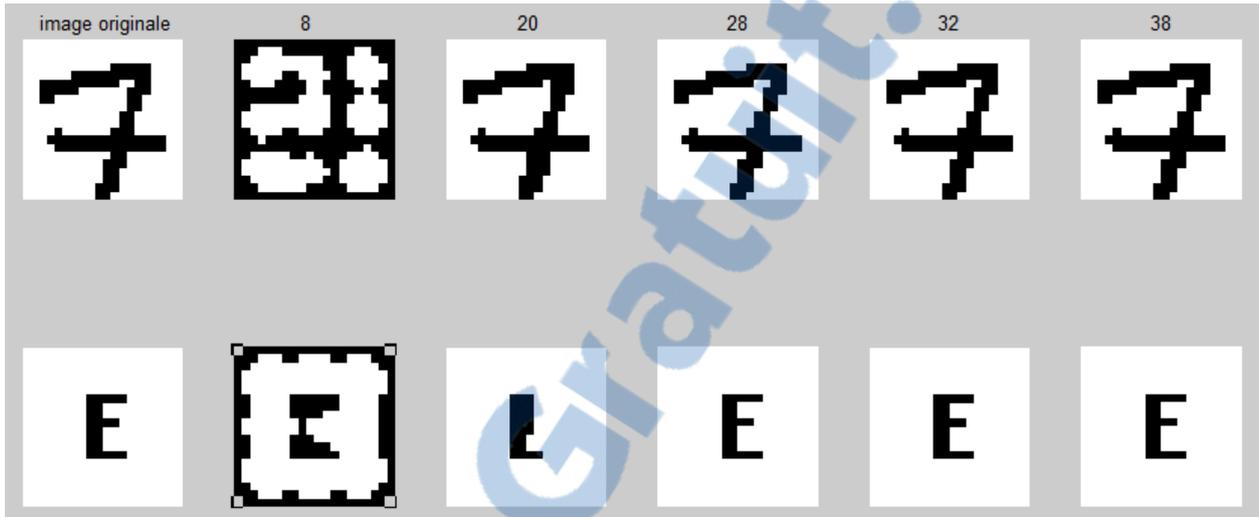


FIGURE 2.14 – reconstruction des images binaires pour ordre $n+m=8$, $n+m=20$, $n+m=28$, $n+m=32$ et $n+m=38$.

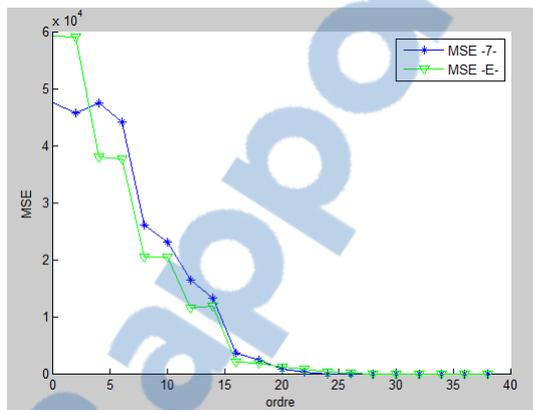


FIGURE 2.15 – la courbe de MSE pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-

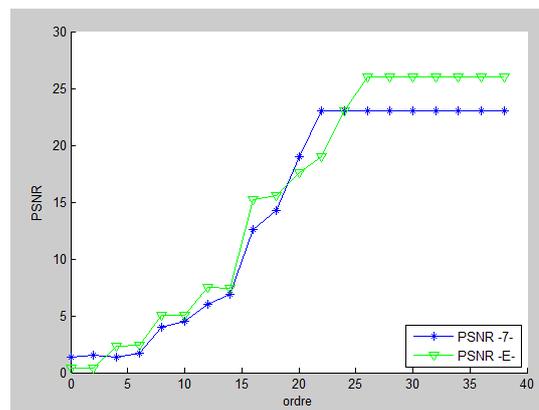


FIGURE 2.16 – la courbe de PSNR pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-

D'après les deux courbes, l'erreur de reconstruction se diminue quand l'ordre dépasse 20, à cet ordre en peut avoir une signification pour l'image reconstruite.

- **Image 2-D niveau de gris -cas sans bruit- :**

Afin de montrer le pouvoir de reconstruction des images niveau de gris 2-D à l'aide de moment de Krawtchouk, les expérimentations suivantes représentent l'évaluation de taux d'erreur de reconstruction via les moments de Krawtchouk.



FIGURE 2.17 – reconstruction des image niveau de gris pour ordre $n+m=20$, $n+m=30$, $n+m=50$, $n+m=70$, $n+m=100$ et $n+m=124$.

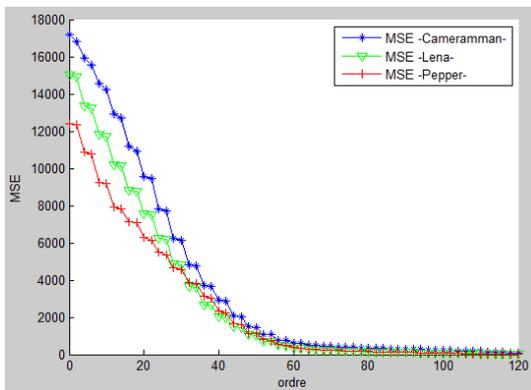


FIGURE 2.18 – la courbe MSE pour différents ordres de reconstruction

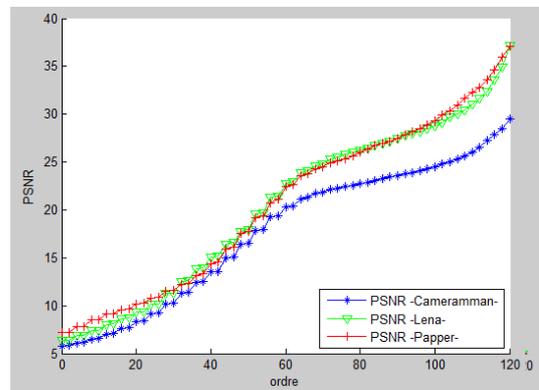


FIGURE 2.19 – la courbe PSNR pour différents ordres de reconstruction

D'après les deux courbes MSE et PSNR présentées en figure (2.18,2.19), l'information reconstruite a une signification dès qu'on atteint l'ordre 60.

- **Image 2-D niveau de gris -cas avec bruit- :**

Dans cette partie nous présentons les différents expérimentations concernant la reconstruction des images bruitées, nous avons choisi un bruit impulsionnel et gaussien de densité 0.02, la reconstruction des images est donnée comme suit :

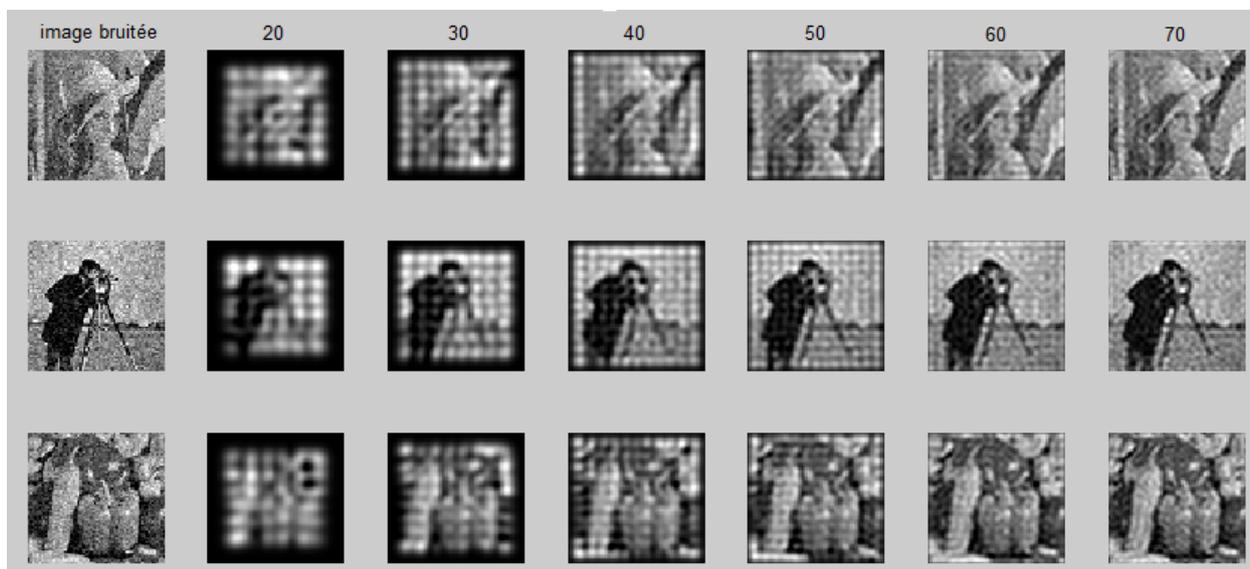


FIGURE 2.20 – la reconstruction des images bruitées avec bruit gaussienne de densité $d=0.02$.

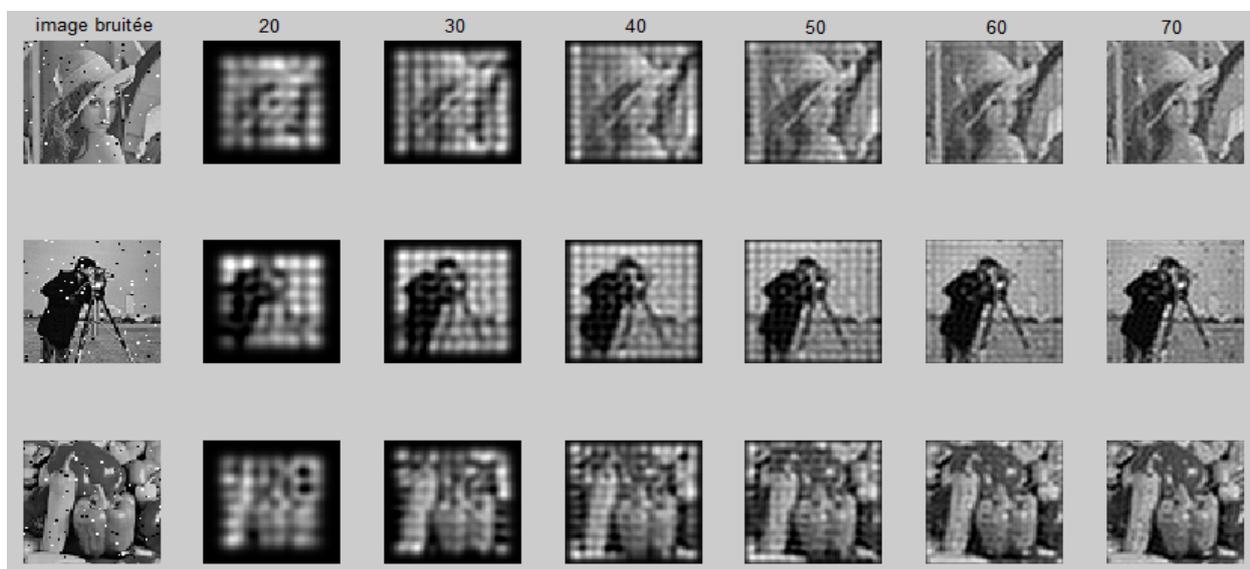


FIGURE 2.21 – la reconstruction des images bruitées avec bruit impulsionnel de densité $d=0.02$.

Afin de visualiser le taux d'erreur de reconstruction au niveau des images bruitées et originales, nous présentons les courbes MSE et PSNR qui décrivent cette variation de taux d'erreur.

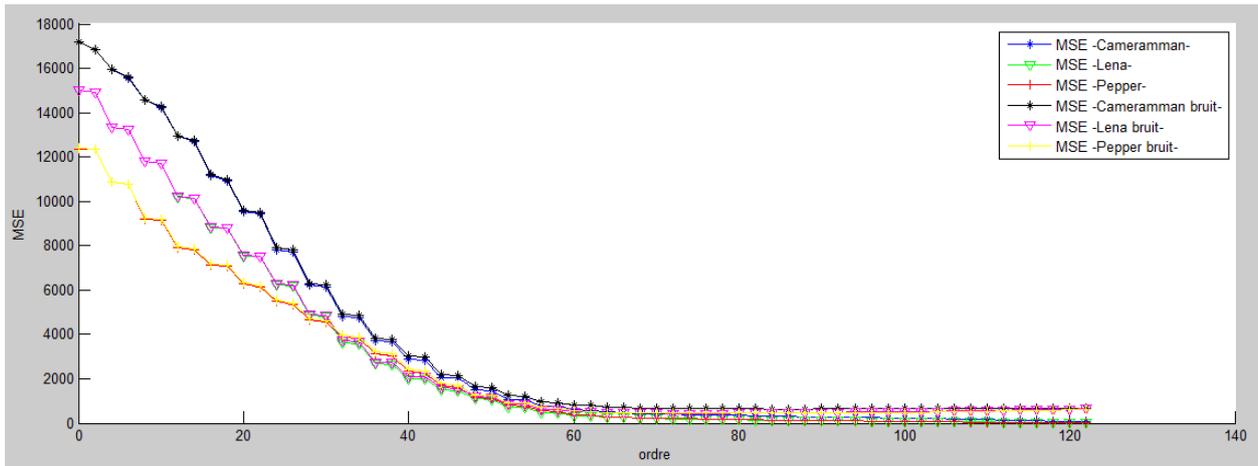


FIGURE 2.22 – le taux d’erreur de reconstruction des reconstruction des images bruitées avec bruit gaussienne de densité $d=0.02$

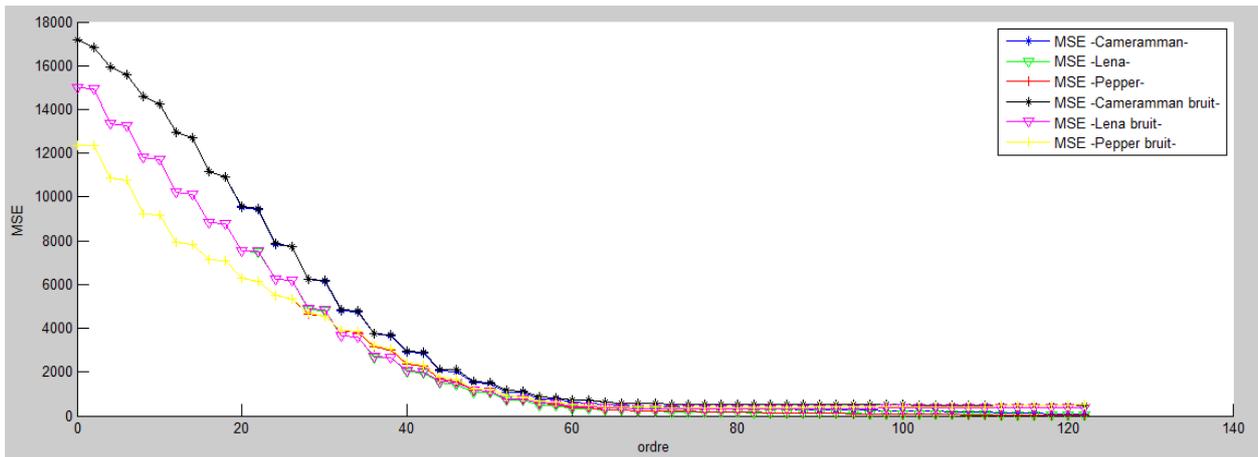


FIGURE 2.23 – taux d’erreur de reconstruction des reconstruction des images bruitées avec bruit impulsif de densité $d=0.02$

Après que nous avons exposé les différentes expérimentations de reconstruction des images 2-D (niveau de gris et binaires) via les moments de Krawtchouk, nous continuons par les expérimentations obtenues au niveau de Moment de Hahn.

2.2.4 Reconstruction des images 2-D à partir de Moments Hahn

En suivant les mêmes démarches , nous présentons les algorithmes de calcul des polynômes de Hahn, Moment de Hahn et la reconstruction d’une image à partir les moments de Hahn, les algorithmes sont donnés comme suit :

Algorithm 7: Calcul des polynômes discrets de Hahn**Input:** N un entier positif exprime la taille de l'image , μ et ν paramètres d'ajustement**Output:** H matrice de polynômes de Hahn normalisés

```
1 for  $i \leftarrow 0$  to  $N - 1$  do
2    $H_0(i) = ((N + \nu - 1)(N - 1) - (2N + \mu + \nu - 2)i) \sqrt{\frac{\rho(i)}{d_1^2}}$  ;
3    $H_1(i) = \sqrt{w(x; p, N - 1)} \frac{(N-1)p-x}{\sqrt{(N-1)p(1-p)}}$  ;
4   for  $n \leftarrow 2$  to  $N - 1$  do
5      $A = \frac{n(2N+\mu+\nu-n)}{\sqrt{(2N+\mu+\nu-2n+1)(2N+\mu+\nu-2n)}}$  ;
6      $B = i - \frac{2(N-1)+\nu-\mu}{4} - \frac{(\mu^2-\nu^2)(2N+\mu+\nu)}{4(2N+\mu+\nu-2n+2)(2N+\mu+\nu-2n)}$  ;
7      $C = \frac{(N-n+1)(N-n+\mu+1)(N-n+\mu+1)(N-n+\mu+\nu+1)}{(2N+\mu+\nu-2n+2)(2N+\mu+\nu-2n+1)}$  ;
8      $AH_n^{(\mu,\nu)}(i) = B\sqrt{\frac{d_{n-1}^2}{d_n^2}}H_{n-1}^{(\mu,\nu)}(i) - C\sqrt{\frac{d_{n-2}^2}{d_n^2}}H_{n-2}^{(\mu,\nu)}(i);$ 
```

Algorithm 8: Calcul des moments discrets de Hahn**Input:** $Nmax < N$ un entier positif exprime l'ordre de reconstruction, f l'image 2-D en de taille $N*N$, h matrice de polynôme**Output:** H matrice de moment

```
1 for  $n \leftarrow 0$  to  $Nmax$  do
2   for  $m \leftarrow 0$  to  $Nmax$  do
3      $sum \leftarrow 0$ 
4     for  $x \leftarrow 0$  to  $N - 1$  do
5       for  $y \leftarrow 0$  to  $N - 1$  do
6          $sum = sum + (h_n^{(\mu,\nu)}(x)h_m^{(\mu,\nu)}(y)f(x, y));$ 
7      $H_{nm} = sum;$ 
```

Algorithm 9: Reconstruction des images 2-D en niveaux de gris**Input:** $Nmax < N$ un entier positif exprime l'ordre de reconstruction , H la matrice des moment, h la matrice de polynômes**Output:** $image_reconstruite$

```
1 for  $x \leftarrow 0$  to  $N$  do
2   for  $y \leftarrow 0$  to  $N$  do
3      $sum \leftarrow 0$ 
4     for  $n \leftarrow 0$  to  $Nmax$  do
5       for  $m \leftarrow 0$  to  $Nmax$  do
6          $sum = sum + (h_n^{(\mu,\nu)}(x)h_m^{(\mu,\nu)}(y)H_{nm});$ 
7      $image\_reconstruite(x, y) = sum;$ 
```

Après la présentation des algorithmes de calcul, nous exposons les différentes expérimentations réalisées au niveau de reconstruction des images 2-D à partir des moments de Hahn, nous admettons que durant toutes les expériences les paramètres μ et ν sont mis à zéro.

- **Image 2-D binaire :**

Dans cette partie nous prenons les images introduites à la figure (2.1), nous présentons les différents ordres de reconstruction, et les courbes qui présentent le taux d'erreur au niveau de reconstruction des images 2-D par rapport à l'image originale.

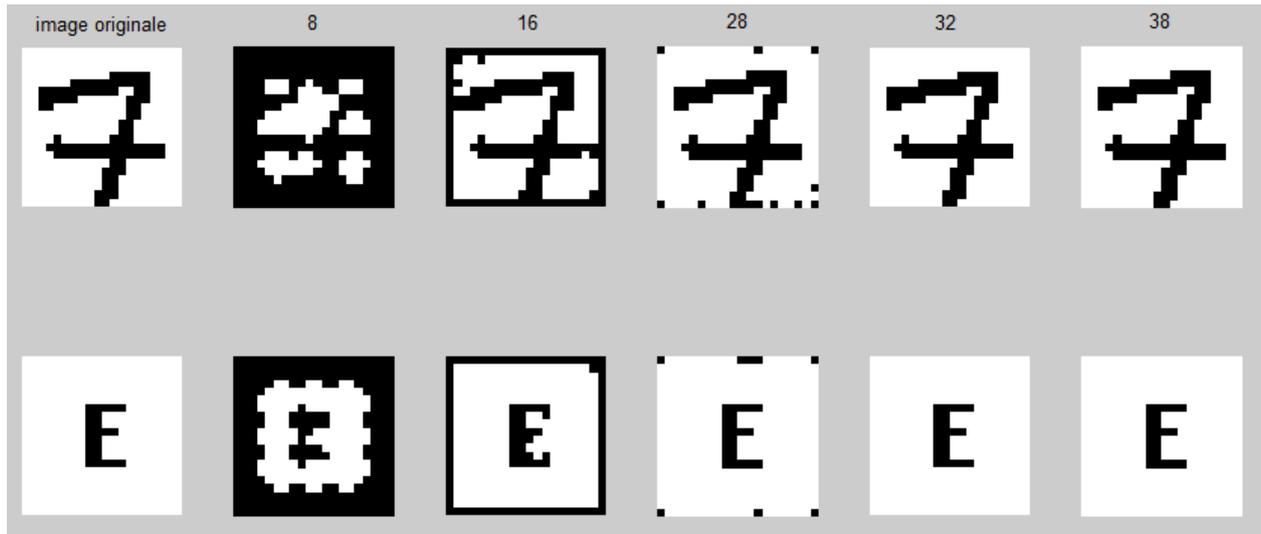


FIGURE 2.24 – reconstruction des images binaires pour ordre $n+m=8$, $n+m=20$, $n+m=28$, $n+m=32$ et $n+m=38$.

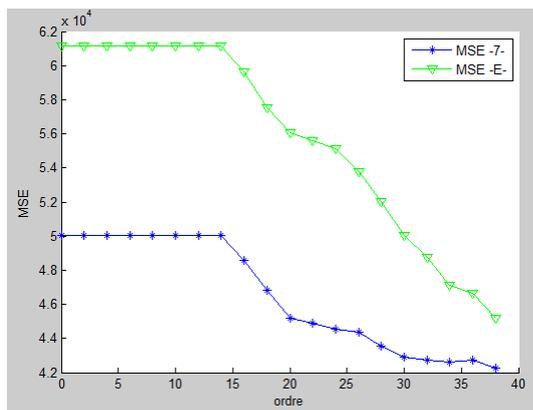


FIGURE 2.25 – la courbe de MSE pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-

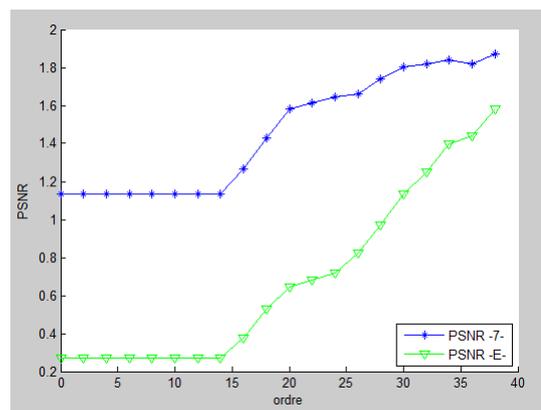


FIGURE 2.26 – la courbe de PSNR pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-

Comme illustre la courbe de MSE et PSNR, l'information n'a pas de sens que lorsqu'on atteint les valeurs d'ordre 16 et plus, ceci est exprimé en reconstruction des images représentées en figure (2.24).

- **Image 2-D niveau de gris -cas sans bruit- :**

Dans l'objectif de montrer le pouvoir de reconstruction des images niveau de gris 2-D à partir des moments de Hahn, nous présentons les expérimentations suivante, qui illustrent l'évaluation de taux d'erreur de reconstruction via les moments de Hahn.



FIGURE 2.27 – reconstruction des images niveau de gris pour ordre $n+m=20$, $n+m=30$, $n+m=50$, $n+m=70$, $n+m=100$ et $n+m=124$.

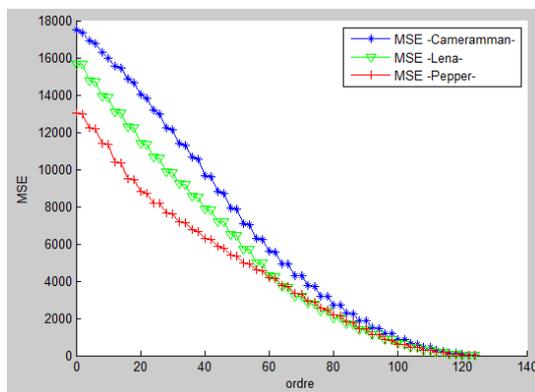


FIGURE 2.28 – courbe MSE pour différents ordres

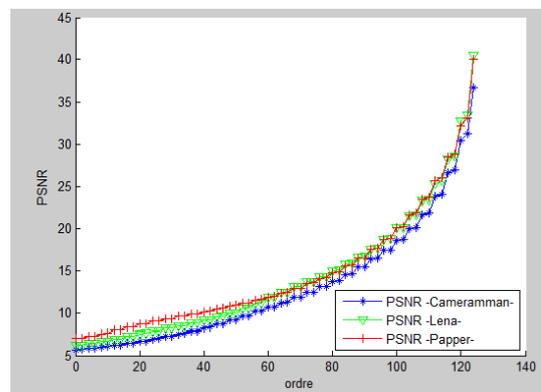


FIGURE 2.29 – courbe PSNR pour différents ordres

Comme interprétation, la reconstruction de l'information n'a pas une signification que lorsqu'on atteint les ordres maximums de reconstruction.

- **Image 2-D niveau de gris -cas avec bruit- :**

Afin d'évaluer la robustesse de reconstruction des images 2-D niveau de gris bruitées à l'aide des moments de Hahn, nous introduisons les différentes expérimentations obtenus avec différent types de bruits, gaussien et impulsionnel.

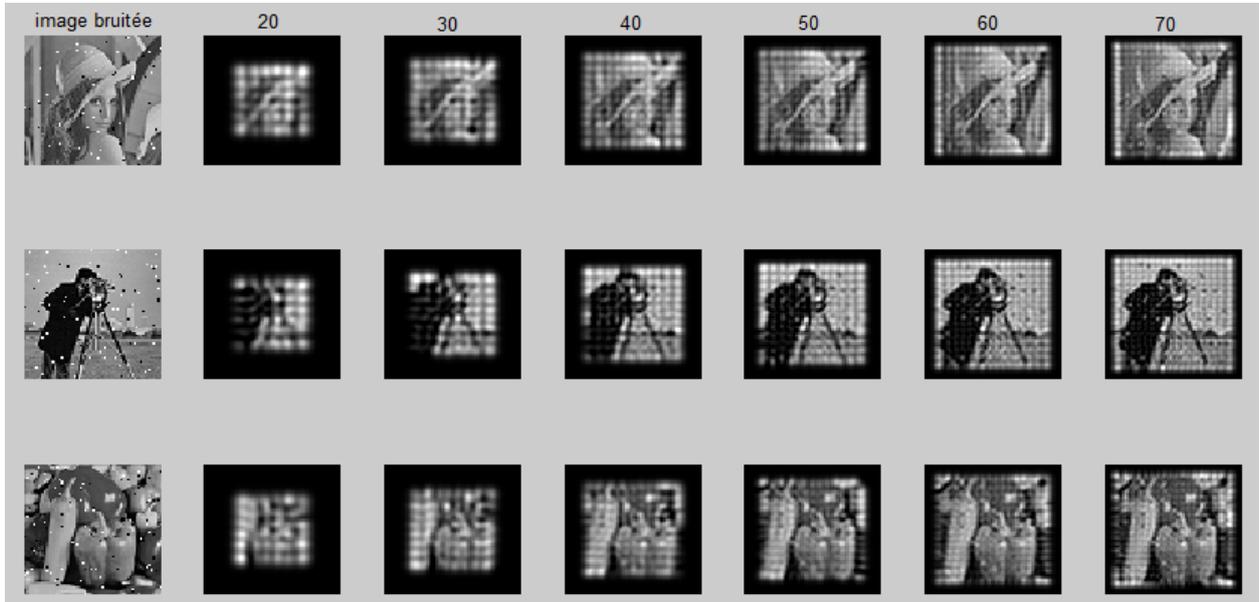


FIGURE 2.30 – reconstruction des images bruité avec bruit impulsionnel de densité $d=0.02$.

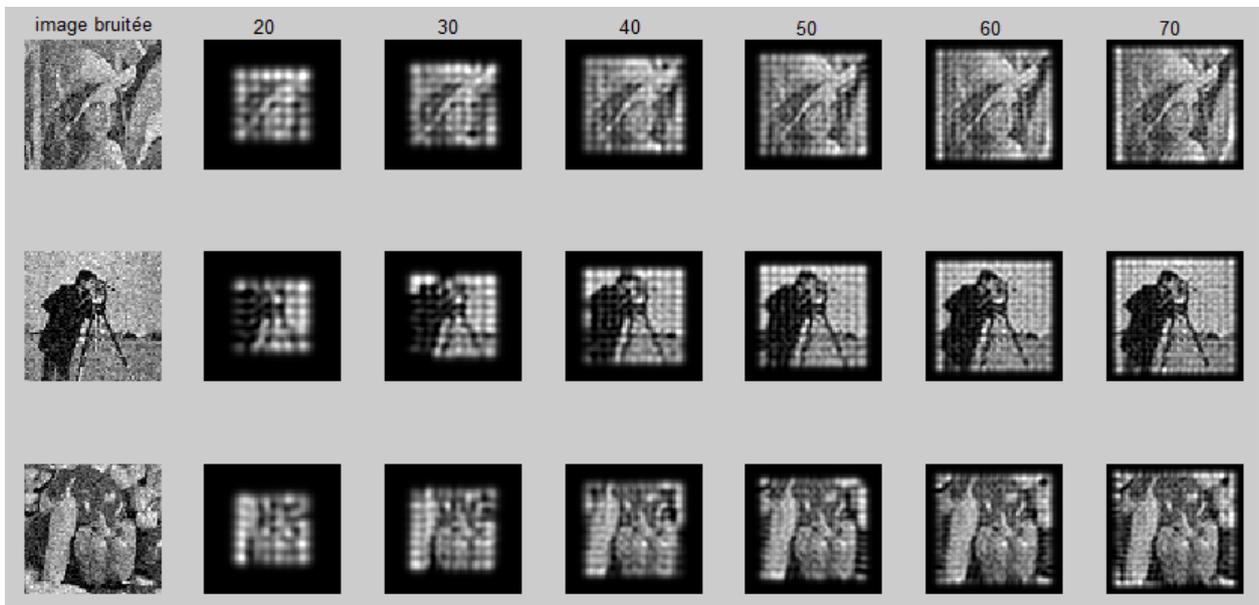


FIGURE 2.31 – reconstruction des images bruité avec bruit gaussien de densité $d=0.02$.

Afin de garantir l'aspect de l'analyse au niveau des images bruitées, nous l'exprimons avec des courbes qui évaluent le taux d'erreur entre l'image originale et bruitée.

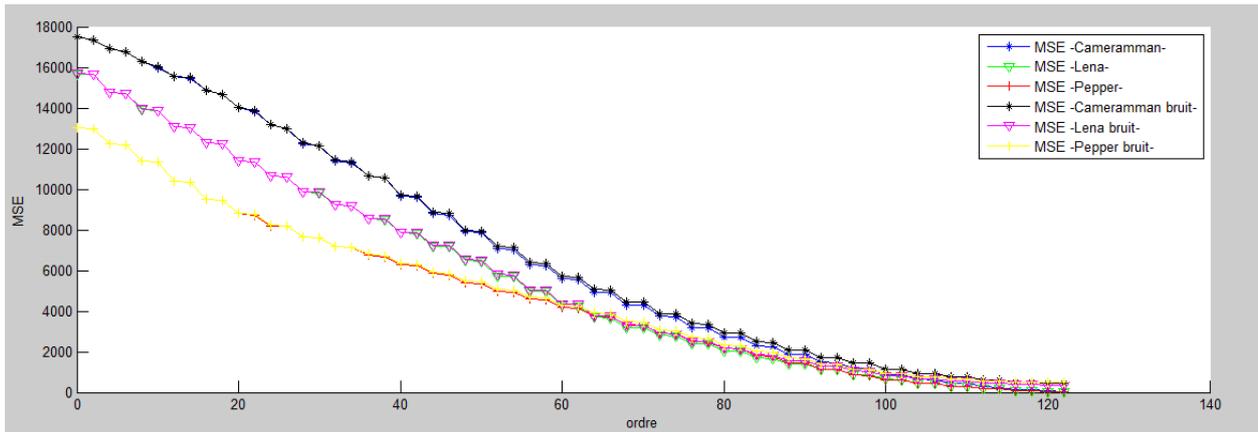


FIGURE 2.32 – reconstruction des images bruitées avec bruit impulsional de densité $d=0.02$.

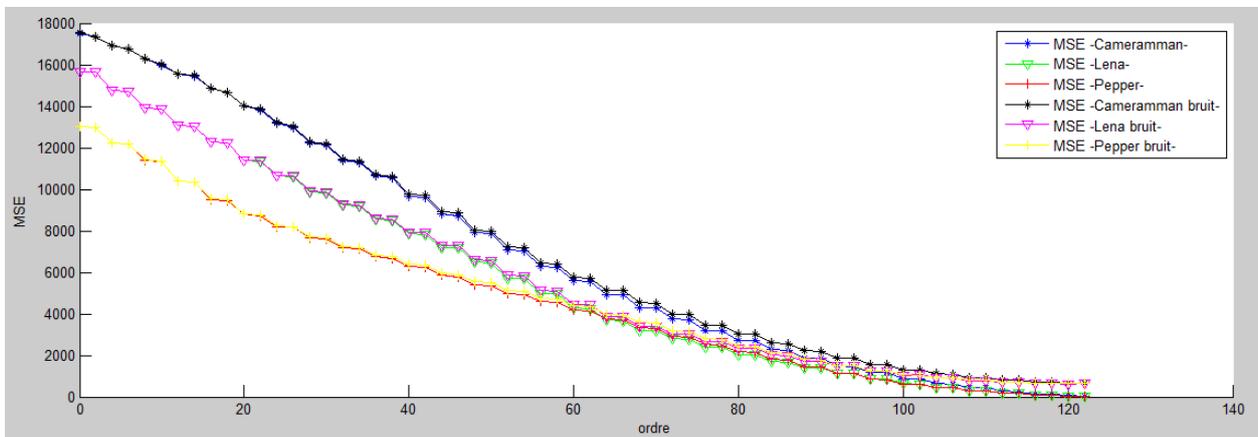


FIGURE 2.33 – reconstruction des images bruitées avec bruit gaussienne de densité $d=0.02$.

Nous finissons cette partie, par la présentation des expérimentations obtenues au niveau de reconstruction des images via les moments de DualHahn.

2.2.5 Reconstruction des images 2-D à partir de Moments de Dual-Hahn

Nous commençons par l'introduction des algorithmes de calcul des polynômes et moments de DualHahn, et l'algorithme de reconstruction des images, en suite nous présentons les différentes expérimentations réalisées à ce niveau. Dans notre étude les paramètres a et c sont mis à zéro et b sera la taille de l'image traitée.

Algorithm 10: Calcul des polynômes discrets de DualHahn**Input:** N un entier positif exprime la taille de l'image , a, b, c paramètres d'ajustement**Output:** W matrice de polynômes de DualHahn normalisés

```

1 for  $i \leftarrow 0$  to  $N - 1$  do
2    $s = s + a$ 
3    $W_0(i) = \left( \frac{\rho(s)(2s+1)}{d_0^2} \right) ;$ 
4    $W_1(i) = \left[ (\rho_1(s-1) - \rho_1(s)) / (\rho(s)(2s+1) \times \sqrt{(\rho(s)(2s+1)) / d_1^2}) \right] ;$ 
5   for  $n \leftarrow 2$  to  $N - 1$  do
6      $A = \frac{1}{n}(s \times (s+1) - ab + ac - bc - (b-a-c-1) \times (2n-1) + 2 \times (n-1)^2) ;$ 
7      $B = \frac{-1}{n} \times (a+c+n-1) \times (b-a-n+1) \times (b-c-n+1) ;$ 
8      $D = \sqrt{\frac{n}{(a+c+n) \times (b-a-n) \times (b-c-n)}} ;$ 
9      $F = \sqrt{\frac{n(n-1)}{(a+c+n) \times (a+c+n-1) \times (b-a-n+1) \times (b-a-n) \times (b-c-n+1) \times (b-c-n)}} ;$ 
10     $W_n(i) = A \times D \times W_{n-1}(i) + B \times F \times W_{n-2}(i) ;$ 
11 Return  $W$ 

```

Algorithm 11: Calcul des moments discrets de DualHahn**Input:** $Nmax < N$ un entier positif exprime l'ordre de reconstruction, f l'image 2-D en de taille $N \times N, w$ matrice de polynôme**Output:** W matrice de moment

```

1 for  $n \leftarrow 0$  to  $Nmax$  do
2   for  $m \leftarrow 0$  to  $Nmax$  do
3      $sum \leftarrow 0$ 
4     for  $x \leftarrow 0$  to  $N - 1$  do
5       for  $y \leftarrow 0$  to  $N - 1$  do
6          $sum = sum + (w_n(x)w_m(y)f(x, y));$ 
7        $H_{nm} = sum;$ 
8 Return  $W$ 

```

Algorithm 12: Reconstruction des images 2-D en niveaux de gris

Input: $Nmax < N$ un entier positif exprime l'ordre de reconstruction , W la matrice des moment, w la matrice de polynômes

Output: $image_reconstruite$

```
1 for  $x \leftarrow 0$  to  $N$  do
2   for  $y \leftarrow 0$  to  $N$  do
3      $sum \leftarrow 0$ 
4     for  $n \leftarrow 0$  to  $Nmax$  do
5       for  $m \leftarrow 0$  to  $Nmax$  do
6          $sum = sum + (w_n(x)w_m(y)W_{nm});$ 
7        $image\_reconstruite(x, y) = sum;$ 
8 Return  $image\_reconstruite$ 
```

- **Images 2-D binaires :**

Dans cette partie nous présentons les différentes réalisations obtenues par la reconstruction des images binaires via les moments de DualHahn, les images données par la figure (2.1) qui sont utilisées dans cette partie :

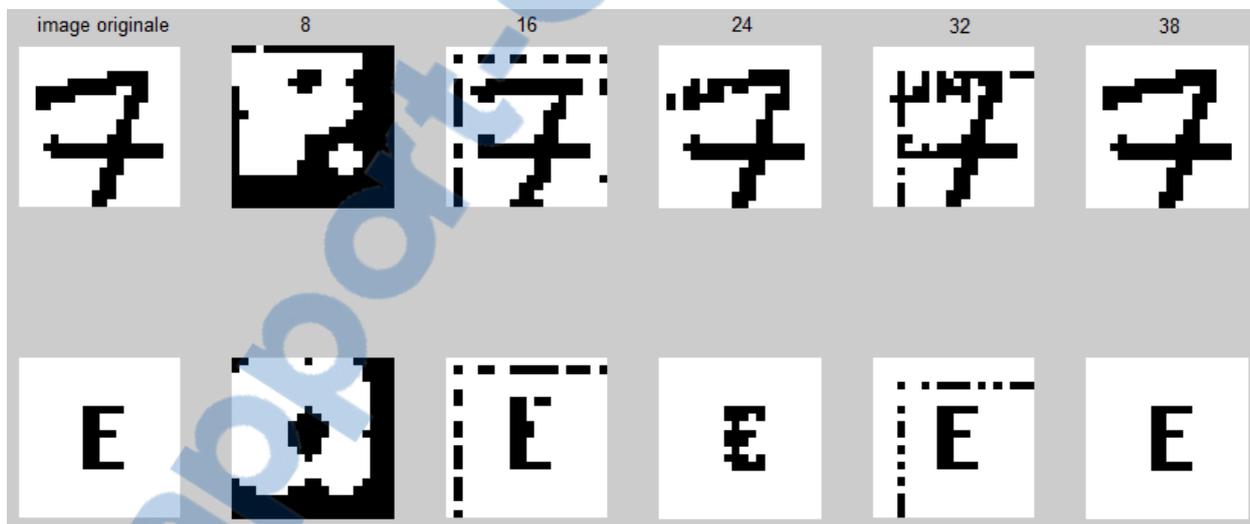


FIGURE 2.34 – reconstruction des images binaires pour les ordres $n+m=8$, $n+m=20$, $n+m=28$, $n+m=32$ et $n+m=38$.

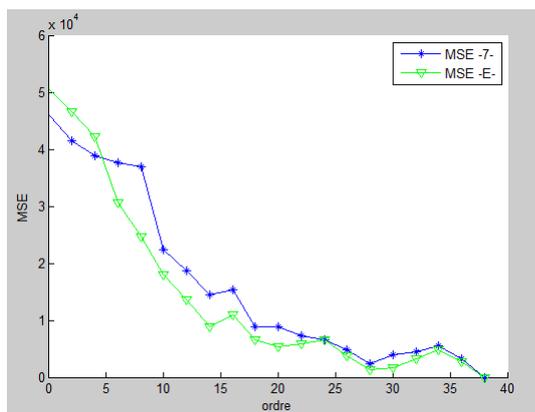


FIGURE 2.35 – la courbe de MSE pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-

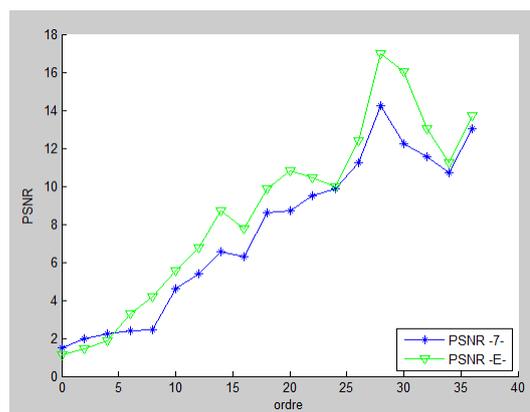


FIGURE 2.36 – la courbe de PSNR pour différents ordres de reconstruction du chiffre -7- et de la lettre -E-

Les deux courbes montrent que l'information devient significative à partir des ordres maximums.

- **Images 2-D niveau de gris -cas sans bruit- :**

Dans l'objectif de montrer le pouvoir de reconstruction des images 2-D niveau de gris à l'aide des moments de DualHahn, les expérimentations au-dessus donnent une interprétation sur la qualité de reconstruction via ces moments, les images montrées au niveau de figure(2.5) celles qui sont utilisées, seront redimensionnées pour remédier au problème de calcul au niveau des polynômes de DualHahn, les nouvelles dimensions des images sont données par $N = M = 50$:

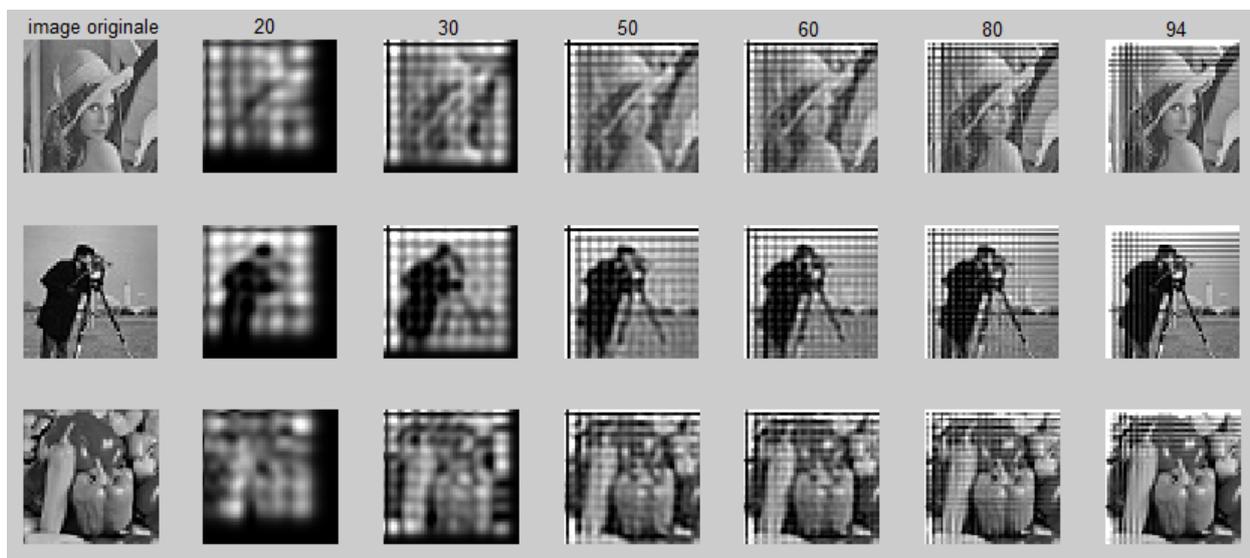


FIGURE 2.37 – reconstruction des images niveau de gris pour ordre $n+m=20$, $n+m=30$, $n+m=50$, $n+m=60$, $n+m=80$ et $n+m=94$.

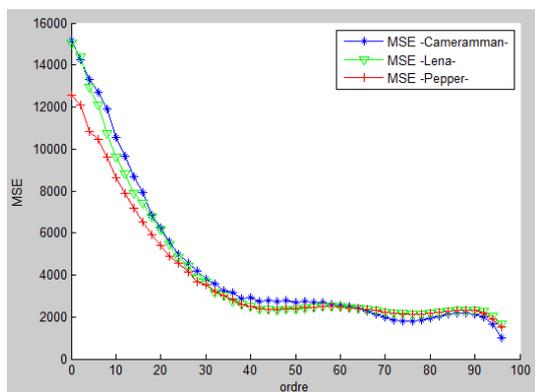


FIGURE 2.38 – courbe MSE pour différents ordres

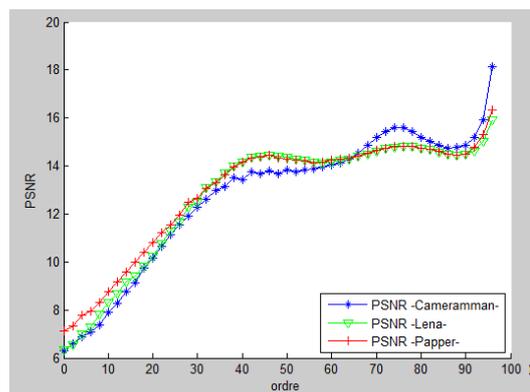


FIGURE 2.39 – courbe PSNR pour différents ordres

A partir des courbes obtenues, le taux d'erreur au niveau de courbe MSE est assez important, et ce qui est visualisé au niveau de la figure de reconstruction (2.37), cela est dû aux erreurs de calcul prévus au niveau de polynôme de DualHahn.

- **Images 2-D niveau de gris -cas bruité- :**

Pour montrer la robustesse des moments de DualHahn au niveau de reconstruction des images bruitées, nous présentons l'étude qui suit dont l'objectif est d'interpréter la qualité de reconstruction à l'aide de ces moments, la densité de bruit durant cette étude est égale à 0.02 :

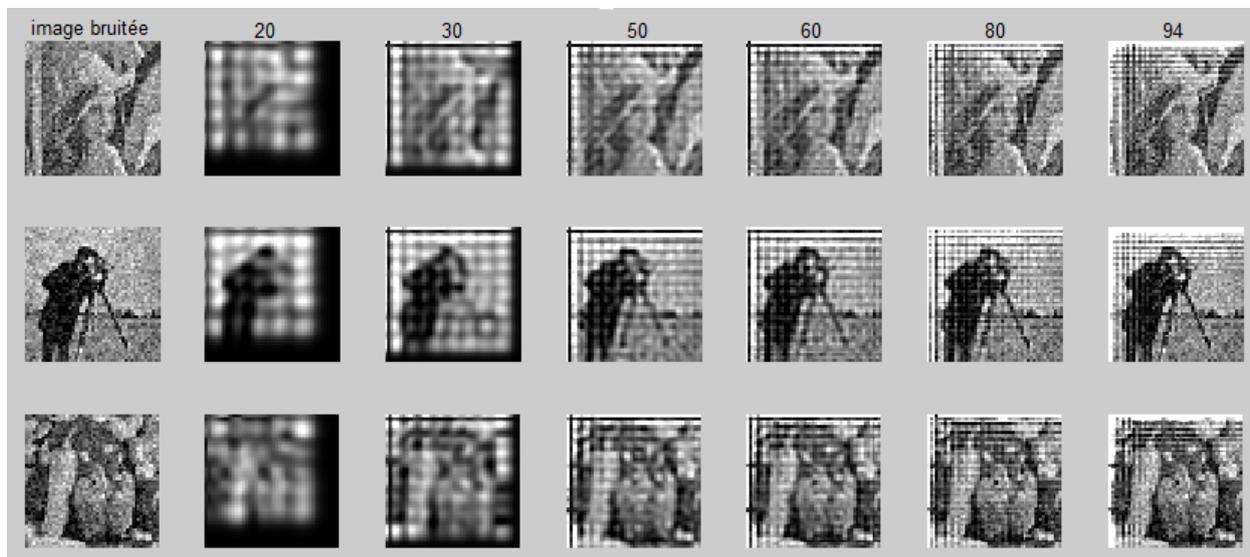


FIGURE 2.40 – reconstruction des images bruitées avec bruit gaussienne de densité $d=0.02$.

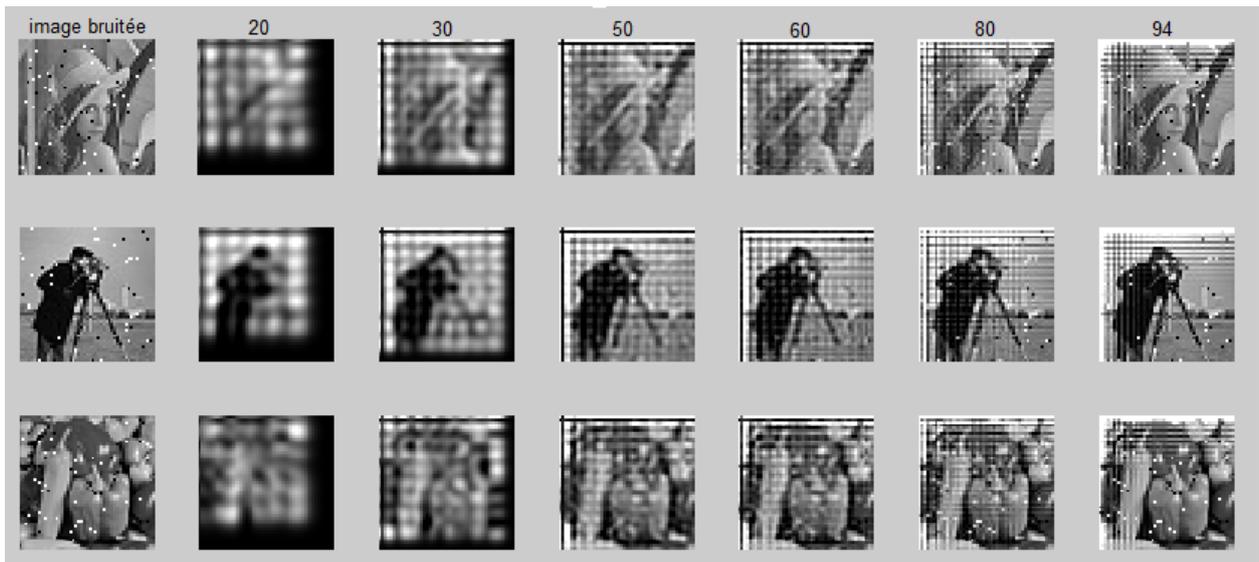


FIGURE 2.41 – reconstruction des images bruitées avec bruit impulsionnel de densité $d=0.02$.

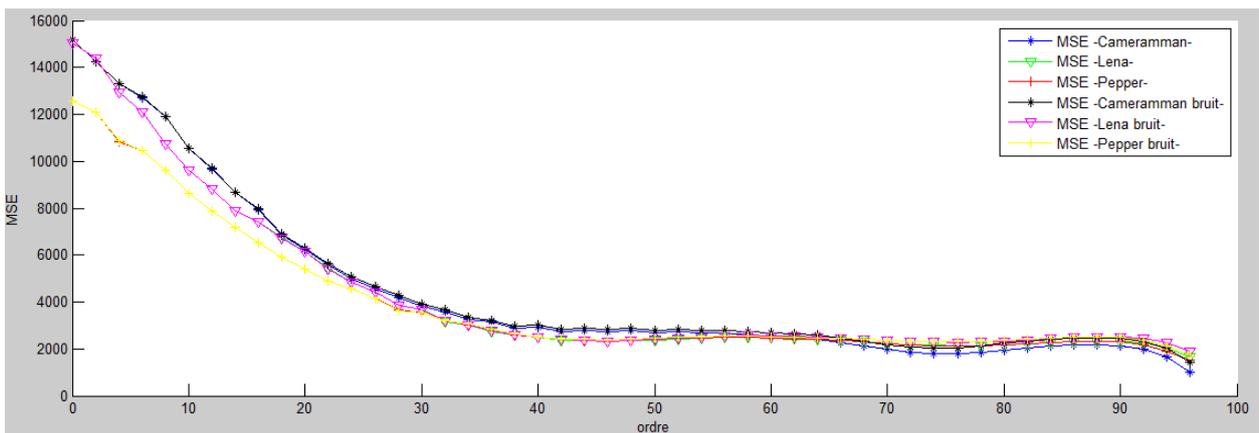


FIGURE 2.42 – reconstruction des images bruitées avec bruit impulsionnel de densité $d=0.02$.

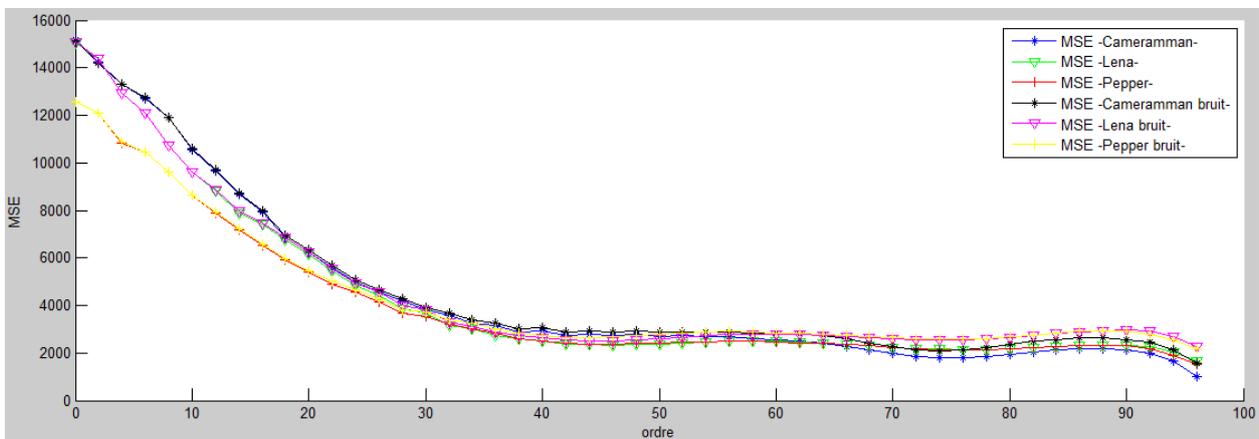


FIGURE 2.43 – reconstruction des images bruité avec bruit gaussienne de densité $d=0.02$.

2.2.6 Comparaison générale

Dans cette partie nous présentons deux courbes MSE et PSNR dont l'objectif est d'évaluer la méthode de reconstruction la plus optimale avec un minimum d'erreur, cette comparaison sera obtenue à partir les moments de Tchebichef, Krawtchouk et Hahn. Nous avons éliminé le moment de DualHahn à cause des erreurs de calcul obtenues durant les expérimentations, les différentes comparaisons sont illustrées au niveau des figures suivantes :

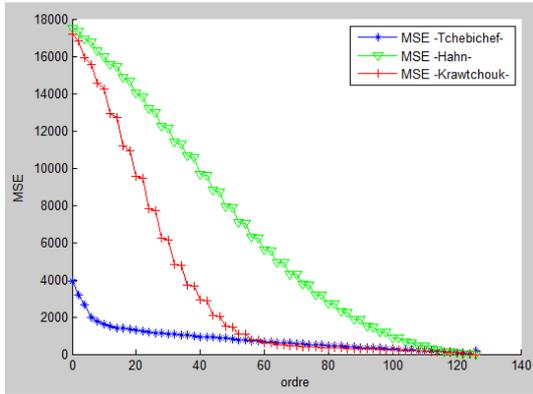


FIGURE 2.44 – courbe MSE pour différents ordres de l'image Cameraman

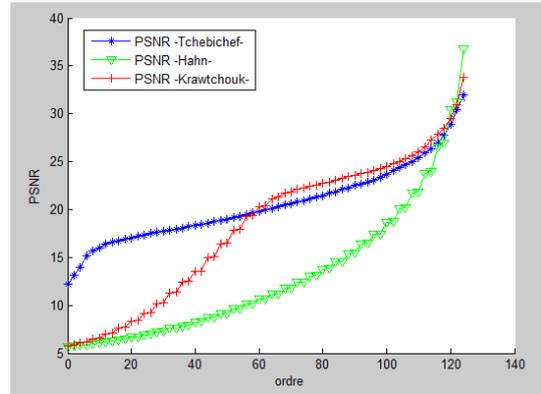


FIGURE 2.45 – courbe PSNR pour différents ordres de l'image Cameraman

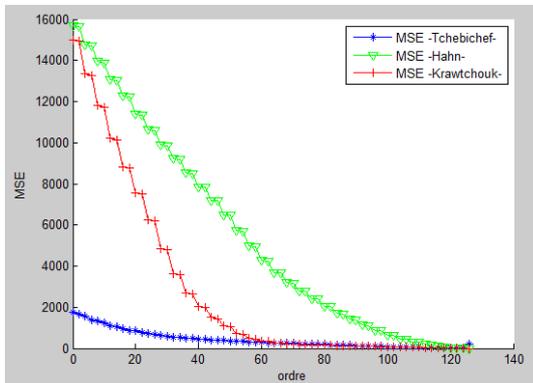


FIGURE 2.46 – courbe MSE pour différents ordres de l'image Lena

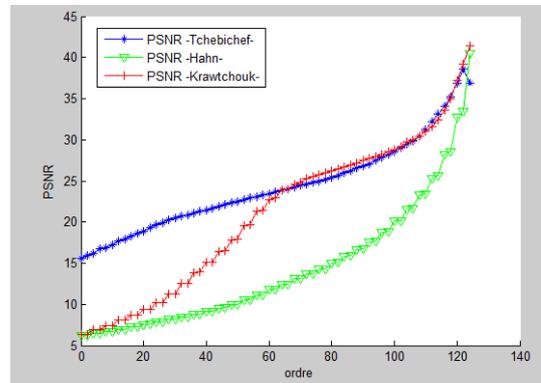


FIGURE 2.47 – courbe PSNR pour différents ordres de l'image Lena

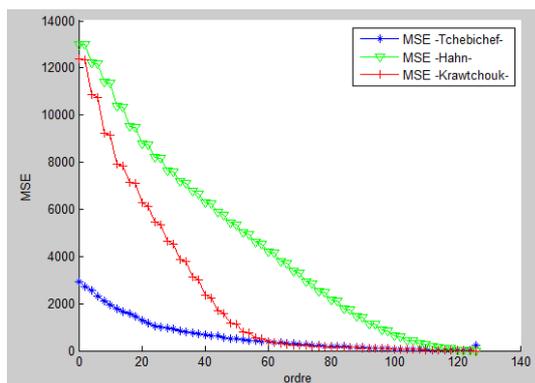


FIGURE 2.48 – courbe MSE pour différents ordres de l'image Papper

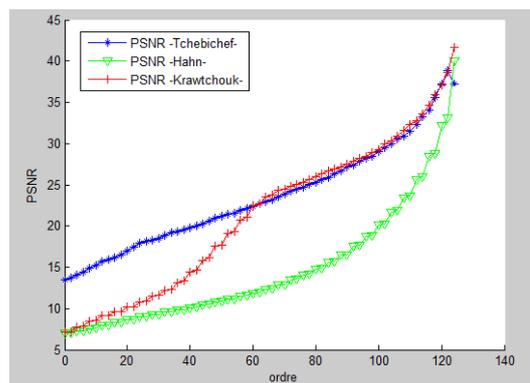


FIGURE 2.49 – courbe PSNR pour différents ordres de l'image Papper

Remarque :

Les courbes présentées au-dessus, permettent d'interpréter que les moments de Krawtchouk sont robustes au niveau de reconstruction, ainsi les moments de Tchebichef, mais ces derniers perdent la signification de l'information dès qu'on atteint les ordres de reconstruction maximum, cela est prévu pour les valeurs des moments d'ordre supérieur (i.e $\exp^{\pm 20}$). Or, les moments de Hahn convergent lentement vers l'image originale. Donc on peut conclure que les moments de Tchebichef et Krawtchouk, sont robustes au niveau de reconstruction.

D'autre part, il est bien connu que la qualité de reconstruction peut être gravement affectée par le bruit de l'image. D'une manière générale, les moments d'ordres plus élevés sont plus sensibles au bruit de l'image[3]. Pour notre travail, les figures illustrées dans les sections précédentes (2.11, 2.12, 2.22, 2.23, 2.32, 2.33, 2.42, 2.43), permettent de visualiser un autre avantage dû aux moments orthogonaux discrets c'est l'élimination de bruit, dans un cadre d'une comparaison générale à partir des huit figures mentionnées en avant, on peut conclure que les moments de Krawtchouk sont les moments moins sensibles aux différents types de bruit, que les autres moments orthogonaux discrets étudiés.

2.3 Conclusion :

Ce chapitre a présenté les différentes implémentations de reconstruction des images 2-D, admises pour les types des moments discrets, de même nous avons présenté une comparaison générale permettant la déduction de performance de chaque moment par rapport aux autres niveaux de reconstruction.

Après cette présentation concernant la reconstruction des images 2-D via les types de moments orthogonaux discrets, le chapitre qui suit, présente l'étude de validité de ces moments en domaine de reconnaissance de forme, on prend l'exemple des images 2-D niveau de gris comme un cas particulier.

Chapitre 3

Application à la reconnaissance de Forme :

3.1 Introduction

La reconnaissance de forme¹ représente une discipline vaste, qui regroupe plusieurs techniques et méthodes visant à identifier une forme au sens informatique à partir des données brute, la reconnaissance de forme peut être effectuée par divers algorithmes d'apprentissage, supervisés ou non-supervisés. Pour notre étude nous nous intéressons aux algorithmes d'apprentissage supervisé, comme cas particulier nous avons opté pour l'algorithme KNN et perceptron multicouche comme exemple pour évaluer la validité des moments de Tchebichef, Krawtchouk et Hahn en domaine de reconnaissance des images 2-D en niveau de gris.

3.2 Méthode de Classification Supervisée

En quelques mots, la classification supervisée est la tâche qui consiste à discriminer des données, de façon supervisée un ensemble de données, de manière que les données d'un même groupe (i.e classe) sont plus proches les unes des autres par rapport au celles d'autres groupes.

En général, on passe par une phase d'apprentissage dont l'objectif est d'apprendre une règle de classification à partir des données par l'expert (i.e classifieur), afin de déterminer les classes appropriées pour chaque donnée d'apprentissage, ensuite on passe à la phase de prédiction de la classe des nouvelles données qui sera basée sur les classes obtenues en phase d'apprentissage.

Comme nous avons énoncé au début, notre objectif est d'évaluer la validité de ces mo-

1. Le mot forme est au sens très général, il ne s'agit pas que de forme géométrique. Les formes ou motifs à reconnaître peuvent être de natures très variées. Il peut s'agir de contenu visuel (code barre, visage, empreinte digitale...) ou sonore (reconnaissance de parole), d'images médicales (rayon X, EEG, IRM...) ou multi-spectrales (images satellitaires) et bien d'autres.

ments obtenus en domaine de reconnaissance de forme, pour cela nous avons opté pour la classification supervisée des images 2-D en niveau de gris comme exemple de reconnaissance de forme, les méthodes suivies pour atteindre cet objectif sont : la méthode de KNN et le perceptron multicouche - Algorithme de rétropropagation -, afin de mieux éclaircir le fonctionnement du système de classification réalisé dans notre travail, le schéma en figure (3.1) résume ce fonctionnement :

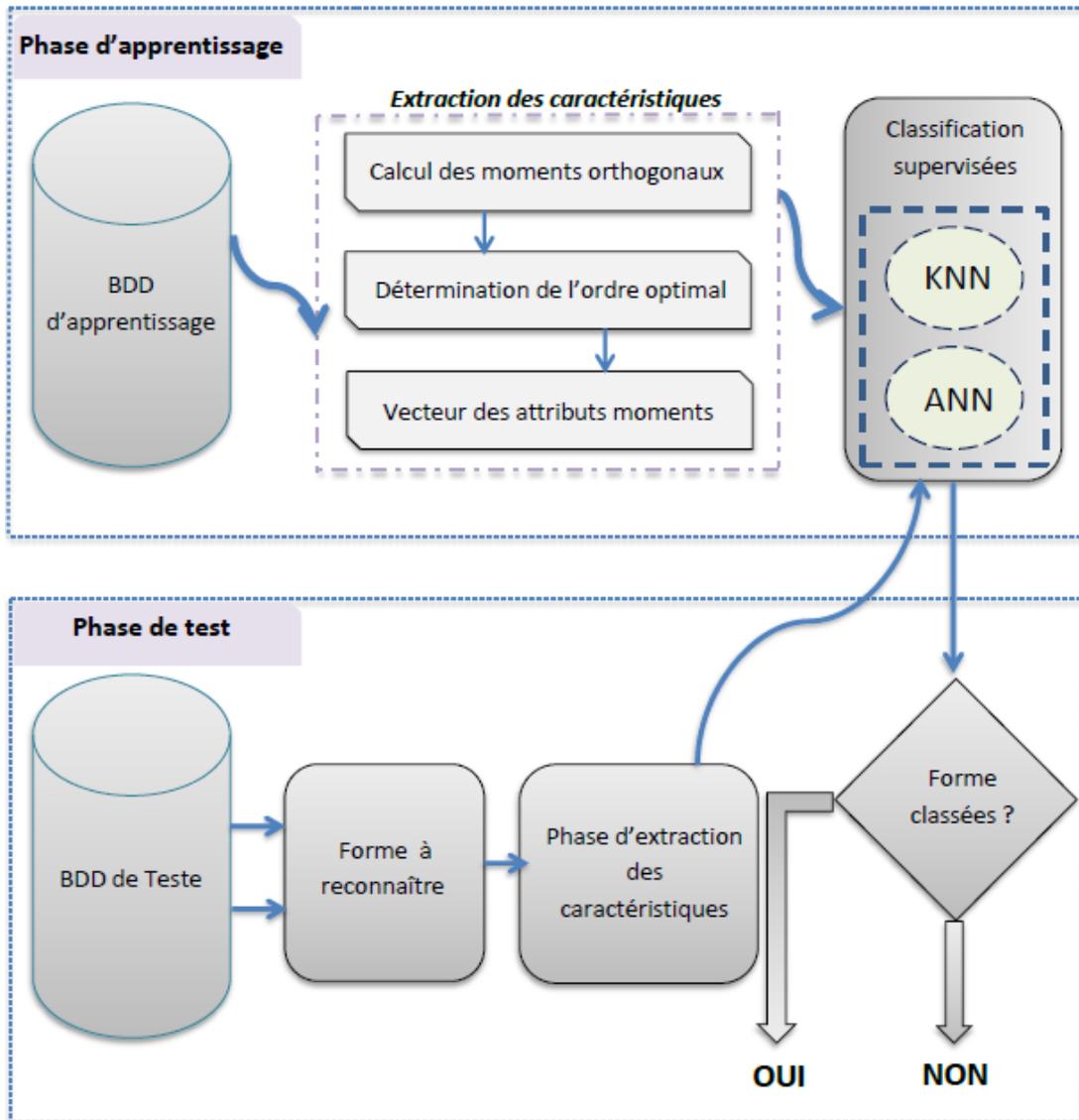


FIGURE 3.1 – système de classification

L'architecture de système de classification proposé au-dessus, est composée de deux étapes importantes, à savoir l'extraction des attributs caractéristiques et les méthodes de classification. La phase d'extraction des caractéristiques sera traitée au niveau de section Résultats et Analyses. Ainsi que les méthodes de classification seront détaillées dans la section suivante.

3.2.1 K-plus proche voisin

Plus connu en anglais sous le nom K-nearest neighbor, cet algorithme réside parmi les plus simples en apprentissage artificiel [9]. Dans le contexte de classifier une nouvelle observation X , l'idée est de chercher les K plus proches voisins, ensuite nous attribuons cette observation à la classe majoritaire obtenue à partir de cette observation. Comme cette méthode est basée sur la notion de plus proche voisin, cette notion sera évaluée en termes de distance. En particulier en \mathbb{R}^n , *Minkowski* a présenté une distance générale entre deux points de \mathbb{R}^n $P1(x_1, x_2, \dots, x_n)$ et $P2(x_1, x_2, \dots, x_n)$, la distance est donnée par :

$$d(P1, P2) = \sqrt[p]{\sum_{i=1}^n |x_i - y_i|^p}. \quad (3.1)$$

Comme un cas particulier pour $p = 1$ on obtient la distance de *Manhattan*, et pour $p=2$ on a la distance *euclidienne*. Dans notre étude la règle de classification admise c'est la distance euclidienne. L'algorithme de KNN revient à implémenter le 1-NN qui est la base de KNN.

- **Algorithme 1-NN**

La méthode de plus proche voisin permet de classer une observation dans la classe d'appartenance de l'observation d'échantillon d'apprentissage qui lui est la plus proche, La détermination de similarité est basée sur la mesure de distance, formellement l'algorithme de plus proche voisin est donné comme suit :

Algorithm 13: Algorithme du plus proche voisin(1-ppv)	
Input:	Données d'apprentissage ; $X^{train} = (x_1^{train}, \dots, x_n^{train})$, classes des données d'apprentissage $Z^{train} = (z_1^{train}, \dots, z_n^{train})$, $X^{test} = (x_1^{test}, \dots, x_m^{test})$
Output:	$Z^{test} = (z_1^{test}, \dots, z_n^{test})$
1	for $i \leftarrow 1$ to m do
2	for $j \leftarrow 1$ to n do
3	calculer la distance euclidienne entre x_i^{test} et x_j^{train}
4	$d_j \leftarrow d(x_i^{test}, x_j^{train})$;
5	Calculer la classe z_i^{test} du i ème exemple qui vaut la classe de son ppv :
6	trouvez l'indice de ppv de x_i^{test} :
7	$ind_ppv_i \leftarrow \operatorname{argmin}_{j=1}^n d_j$
8	trouvez la classe de ppv de x_i^{test} (qui est $x_{ind_ppv_i}^{test}$)
9	$z_i^{test} = x_{ind_ppv_i}^{train}$
10	Return classe des données de test $z^{test} = (z_1^{test}, \dots, z_n^{test})$

- **Algorithme KNN**

Cette méthode est largement utilisée en classification, c'est une extension d'algorithme 1-NN. La plus proche observation n'est plus la seule utilisée pour la classification. Nous

utilisons les k plus proches observations. Ainsi la décision est en faveur de la classe majoritaire représentée par les k voisins.

Soit k_r , le nombre d'observation issu de groupe des plus proches voisins appartenant à la classe r :

$$\sum_{r=1}^c k_r = k \quad (3.2)$$

Ainsi une nouvelle observation est prédite dans la classe résultante est obtenue par :

$$l = \max_r(k_r) \quad (3.3)$$

Ceci évite que la classe prédite ne soit déterminée qu'à partir d'une seule observation, donc le degré de prédiction est obtenu par le paramètre k , l'algorithme qui implémente cette méthode est donné comme suit :

Algorithm 14: Algorithme des k plus proches voisins	
Input: Données d'apprentissage ; $X^{train} = (x_1^{train}, \dots, x_n^{train})$, classes des données d'apprentissage $Z^{train} = (z_1^{train}, \dots, z_n^{train})$, $X^{test} = (x_1^{test}, \dots, x_m^{test})$, nombre des ppv K	
Output: $Z^{test} = (z_1^{test}, \dots, z_n^{test})$	
1	for $i \leftarrow 1$ to m do
2	for $j \leftarrow 1$ to n do
3	calculer la distance euclidienne entre x_i^{test} et x_j^{train}
4	$d_j \leftarrow d(x_i^{test}, x_j^{train})$;
5	Calculer la classe z_i^{test} du i ème exemple qui vaut la classe de son ppv :
6	/* trouver les $K - ppv$ de x_i^{test} */
7	Trier les distances d_j selon un ordre croissant pour $j=1, \dots, n$.
8	Récupérer en même temps les indices $IndVoisins$ avant le tri des d_j
9	Récupérer les classes des K premiers ppv à partir des indices $IndVoisins$ et trouvez la classe majoritaire.
10	$C_k \leftarrow 0 (k = 1, \dots, n)$
11	for $j \leftarrow 1$ to n do
12	$ind_voisin_k \leftarrow IndVoisins_k$
13	$h \leftarrow z_n^{ind_voisin_k}$
14	$C_h = C_h + 1$
15	/* trouvez la classe du ppv de x_i^{test}
16	(la classe majoritaire de celle de ses K -ppv)*/
17	$z_i^{test} = \operatorname{argmax}_{k=1}^K C_k$
18	Return classe des données de test $z^{test} = (z_1^{test}, \dots, z_n^{test})$

Remarque :

Le choix de paramètre K réside le problème majeur en algorithme KNN, pour notre cas le choix est réalisé par la variation de K jusqu'à l'obtention d'erreur minimal.

3.2.2 Le Perceptron Multicouche

Le perceptron multicouche est un réseau orienté de neurones artificiels organisé en couches et où l'information voyage dans un seul sens, de la couche d'entrée vers la couche de sortie. La figure (3.2) donne l'exemple d'un réseau contenant une couche d'entrée, deux couches cachées et une couche de sortie. La couche d'entrée représente toujours une couche virtuelle associée aux entrées du système. Elle ne contient aucun neurone. Les couches suivantes sont des couches de neurones. Dans l'exemple illustré, il y a trois entrées, quatre neurones sur la première couche cachée, trois neurones sur la deuxième et quatre neurones sur la couche de sortie. Les sorties des neurones de la dernière couche correspondent toujours aux sorties du système. Dans le cas général, un perceptron multicouche peut posséder un nombre de couches quelconque et un nombre de neurones (ou d'entrées) par couche également quelconque [10].

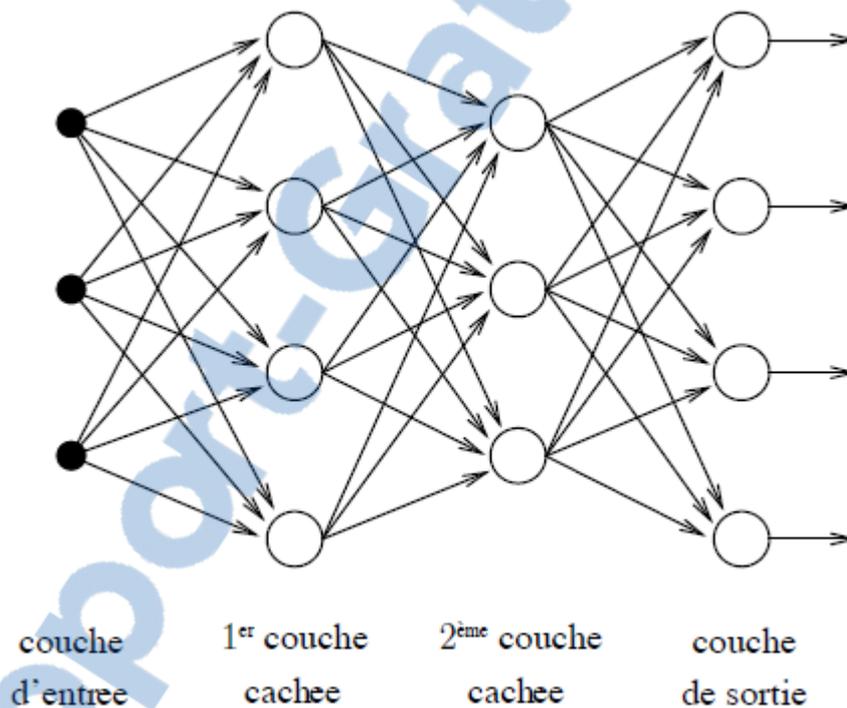


FIGURE 3.2 – Exemple d'un réseau de type perceptron multicouche

Les neurones sont reliés entre eux par des connexions pondérées. Ce sont les poids de ces connexions qui gouvernent le fonctionnement du réseau et “programment” une application de l'espace des entrées vers l'espace des sorties à l'aide d'une transformation non linéaire. La création d'un perceptron multicouche pour résoudre un problème donné passe donc par l'inférence de la meilleure application possible telle que définie par un ensemble de données d'apprentissage constituées de paires de vecteurs d'entrées et de sorties désirées. Cette inférence peut se faire, entre autre, par l'algorithme dit de rétropropagation.

- **Algorithme de rétropropagation**

soit le couple $(\vec{x}(n), \vec{d}(n))$ désignant la n^e donnée d'entraînement du réseau où :

$$\vec{x}(n) = \langle x_1(n), \dots, x_p(n) \rangle \text{ et } \vec{d}(n) = \langle d_1(n), \dots, d_q(n) \rangle \quad (3.4)$$

Ces deux vecteurs correspondent respectivement aux p entrées et aux q sorties désirées du système. L'algorithme de rétropropagation consiste alors à réévaluer l'erreur entre les sorties désirées $\vec{d}(n)$ et les sorties observées par le système $\vec{y}(n)$.

$$\vec{y}(n) = \langle y_1(n), \dots, y_q(n) \rangle \quad (3.5)$$

- **Cas de couche de sortie**

Afin d'évaluer l'erreur observée entre les sorties de système et la sortie désirée, l'équation (1.6) représente l'erreur observée pour le neurone de sortie j et la donnée d'entraînement n :

$$e_j(n) = d_j(n) - y_j(n) \quad (3.6)$$

Cette équation permet d'observer l'erreur dans chaque neurone, or dans un système de neurone, le but est minimiser la somme des erreurs de toutes les neurones de sortie.

Dans notre étude l'indice n représente les données d'entrées, l'indice j représente toujours le neurone pour lequel on veut adapter les poids. Soit $E(n)$ la somme des erreurs quadratiques observées sur l'ensemble C des neurones de sortie :

$$E(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n). \quad (3.7)$$

La sortie $y_j(n)$ du neurone j est définie par :

$$y_j(n) = \varphi[v_j(n)] = \varphi \left[\sum_{i=0}^r w_{ji}(n) y_i(n) \right] \quad (3.8)$$

Où $\varphi[.]$ est la fonction d'activation du neurone, $v_j(n)$ est la somme pondérée des entrées du neurone j , w_{ji} est le poids de la connexion entre le neurone i de la couche précédente et le neurone j de la couche courante et $y_i(n)$ est la sortie du neurone i . On suppose ici que la couche précédente dispose de r neurones numérotés de 1 à r , que le poids $w_{j0}(n)$ correspond au biais du neurone j et que l'entrée $y_0(n) = -1$, Dans notre étude l'indice i représentera toujours un neurone sur la couche précédente par rapport au neurone j , on suppose que cette couche contient r neurones. La figure (1.3) résume ces différentes données :

Pour corriger l'erreur observée, il s'agit de modifier le poids $w_{ji}(n)$ cette modification est obtenue par le calcul de gradient $\frac{\partial E(n)}{\partial w_{ji}(n)}$.

Par la règle de chaînage des dérivées partielles, qui nous dit que $\frac{\partial f(x)}{\partial y} = \frac{\partial f(y)}{\partial y} \cdot \frac{\partial y}{\partial x}$, on obtient :

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = \frac{\partial E(n)}{\partial e_j(n)} \cdot \frac{\partial e_j(n)}{\partial y_j(n)} \cdot \frac{\partial y_j(n)}{\partial v_j(n)} \cdot \frac{\partial v_j(n)}{\partial w_{ji}(n)} \quad (3.9)$$

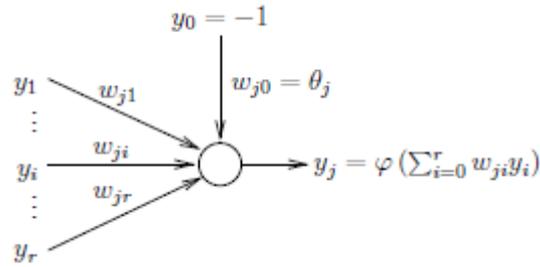


FIGURE 3.3 – Modèle de neurone j

et on exprime la variation de poids $\Delta w_{ji}(n)$ sous la forme :

$$\Delta w_{ji}(n) = -\eta \frac{\partial E(n)}{\partial w_{ji}(n)} \quad (3.10)$$

avec $0 \leq \eta \leq 1$ représente un taux d'apprentissage ou gain d'algorithme.

Évaluons maintenant chacun des termes du gradient :

$$\begin{aligned} \frac{\partial E(n)}{\partial e_j(n)} &= \frac{\partial \left[\frac{1}{2} \sum_{k \in C} e_j^2(n) \right]}{\partial e_j(n)} \\ &= \frac{1}{2} \cdot \frac{\partial e_j^2(n)}{\partial e_j(n)} \\ &= e_j(n) \end{aligned}$$

$$\begin{aligned} \frac{\partial e_j(n)}{\partial y_j(n)} &= \frac{\partial [d_j(n) - y_j(n)]}{\partial y_j(n)} \\ &= -1 \end{aligned}$$

$$\begin{aligned} \frac{\partial Y_j(n)}{\partial v_j(n)} &= \frac{\partial \left[\frac{1}{1 + \exp^{-v_j(n)}} \right]}{\partial v_j(n)} \\ &= \frac{\exp^{-v_j(n)}}{[1 + \exp^{-v_j(n)}]^2} \\ &= y_j(n) \left[\frac{\exp^{-v_j(n)}}{1 + \exp^{-v_j(n)}} \right] \\ &= y_j(n) \left[\frac{\exp^{-v_j(n)+1}}{1 + \exp^{-v_j(n)}} - \frac{1}{1 + \exp^{-v_j(n)}} \right] \\ &= y_j(n) [1 - y_j(n)] \end{aligned}$$

Et finalement

$$\begin{aligned}\frac{\partial v_j(n)}{\partial w_{ji}(n)} &= \frac{\partial [\sum_{l=0}^r w_{jl}(n) y_l(n)]}{\partial w_{ji}(n)} \\ &= \frac{\partial [w_{ji}(n) y_i(n)]}{\partial w_{ji}(n)} \\ &= y_i(n)\end{aligned}$$

Nous obtenons donc

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = -e_j(n) y_j(n) [1 - y_j(n)] y_i(n) \quad (3.11)$$

et la règle "delta" pour la couche de sortie s'exprime par :

$$\Delta w_{ji}(n) = -\eta \frac{\partial E(n)}{\partial w_{ji}(n)} = \eta \delta_j(n) y_i(n) \quad (3.12)$$

avec

$$\delta_j(n) = e_j(n) y_j(n) [1 - y_j(n)] \quad (3.13)$$

qui correspond à ce qu'on l'appelle le gradient local.

A ce stade nous avons traité seulement le cas d'une couche de sortie, il nous reste le cas de traiter l'adaptation des poids pour une couche cachée.

- **Cas de couche cachée**

Considérons maintenant le cas des neurones sur la dernière couche cachée (le cas des autres couches cachées est semblable).

– Le variable n désignera toujours une donnée d'entraînement c'est-à-dire un couple de vecteurs d'entrées et de sorties désirées.

– L'objectif sera toujours d'adapter les poids de la couche courante en minimisant la somme des erreurs sur les neurones de la couche de sortie.

– Les indices i et j désigneront respectivement (comme précédemment) un neurone sur la couche précédente et un neurone sur la couche courante.

– L'indice k servira maintenant à désigner un neurone sur la couche suivante.

Reprenons l'expression de la dérivée partielle de l'erreur totale $E(n)$ par rapport à w_{ji} mais en ne dérivant plus par rapport à l'erreur $e_j(n)$ car celle-ci est maintenant inconnue :

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = \frac{\partial E(n)}{\partial y_j(n)} \cdot \frac{\partial y_j(n)}{\partial v_j(n)} \cdot \frac{\partial v_j(n)}{\partial w_{ji}(n)} \quad (3.14)$$

Le résultat obtenu pour le deuxième et dernier terme au niveau de la première partie reste validé, il nous reste de calculer le premier terme :

$$\frac{\partial E(n)}{\partial y_j(n)} = \frac{\partial [\frac{1}{2} \sum_{k \in C} e_k^2(n)]}{\partial y_j(n)} \quad (3.15)$$

Le problème est que tous les $e_k(k)$ dans la somme ci-dessus dépendent de $h_j(n)$, pour remédier ce problème on réécrit la formule de la façon suivante :

$$\begin{aligned}
\frac{\partial E(n)}{\partial y_j(n)} &= \sum_{k \in C} \left[e_k(n) \frac{\partial e_k(n)}{\partial y_j(n)} \right] \\
&= \sum_{k \in C} \left[e_k(n) \cdot \frac{\partial e_k(n)}{\partial v_k(n)} \cdot \frac{\partial v_k(n)}{\partial y_j(n)} \right] \\
&= \sum_{k \in C} \left[e_k(n) \frac{\partial [d_k(n) - \varphi(v_k(n))]}{\partial v_k(n)} \cdot \frac{\partial [\sum_l w_{kl}(n) y_l(n)]}{\partial y_j(n)} \right] \\
&= \sum_{k \in C} [e_k(n) (-y_k(n) [1 - y_k(n)]) \cdot w_{kj}]
\end{aligned}$$

Et en substituant l'équation(1.13) on obtient :

$$\frac{\partial E(n)}{\partial y_j(n)} = - \sum_{k \in C} \delta_k(n) w_{kj}(n) \quad (3.16)$$

En substituant l'équation (1.16) dans l'équation (1.14), on obtient :

$$\frac{\partial E(n)}{\partial w_{ji}(n)} = -y_j(n) [1 - y_j(n)] \left[\sum_{k \in C} \delta_k(n) w_{kj}(n) \right] y_i(n) \quad (3.17)$$

et

$$\Delta w_{ji}(n) = -\eta \frac{\partial E(n)}{\partial w_{ji}(n)} = \eta \delta_j(n) y_i(n) \quad (3.18)$$

avec

$$\delta_j(n) = y_j(n) [1 - y_j(n)] \sum_{k \in C} \delta_k(n) w_{kj}(n) \quad (3.19)$$

Remarque :

Il faut noter que pour la première couche cachée, il n'y a pas de couche précédente, pour cela il faut substituer la variable $y_i(n)$ par l'entrée $x_i(n)$.

Après la présentation des deux méthodes de classification requises dans notre travail, nous présentons la partie suivante concernant l'évaluation de validité des moments obtenus à partir des images niveau de gris 2-D.

3.3 Résultats et Analyse

3.3.1 Extraction des caractéristiques

Avant de présenter les résultats obtenus au niveau des classifieurs proposés, il y en a une phase importante en système de classification proposée par la figure (3.1), on parle de phase

d'extraction des données caractéristiques. Comme nous avons introduit en première partie de ce document, les moments sont des projections de l'image 2-D sur une base polynomiale. Parmi les problématique proposées par le calcul des moments c'est l'obtention d'ordre optimal duquel on peut avoir une information significative. Pour notre étude nous avons proposé l'extraction d'ordre optimal à partir de la courbe MSE, d'une façon expérimentale, nous avons pris quelques exemples de BD-MNIST d'apprentissage des différents chiffres et nous avons opté pour l'ordre $n + m = 14$ ($n = 7$ et $m = 7$) comme ordre optimal. Pour la seconde base de donnée de YALE Face nous avons opté pour l'ordre optimal $n + m = 126$ ($n = 63$ et $m = 63$) à partir duquel l'information aura une signification.

3.3.2 Reconnaissance des chiffres manuscrits - BD MNIST -

La base de données de MNIST des chiffres manuscrits [20], disponible à partir de <http://yann.lecun.com/exdb/mnist/>, dispose d'un ensemble de formation de 60 000 exemples, et un ensemble de 10 000 exemples de test. Il s'agit d'un sous-ensemble d'un ensemble plus vaste disponible à partir du NIST. Les chiffres sont de taille normalisée et centrée sur une image de taille fixe.

C'est une bonne base de données pour les personnes qui veulent essayer des techniques d'apprentissage et les méthodes de reconnaissance des formes sur des données du monde réel tout en dépensant un minimum d'efforts sur le prétraitement. Cette base de données est celle qui sera admise dans notre évaluation de classification basée sur la notion des moments discrets.

Les deux figures (3.4 et 3.5) au-dessous présente successivement des exemples d'apprentissage et de test, pris de BD-Mnist.

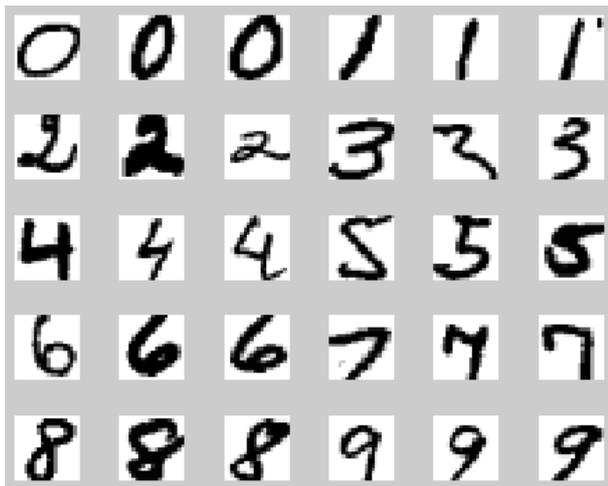


FIGURE 3.4 – Exemple d'apprentissage



FIGURE 3.5 – Exemple de test

- **Résultat obtenu en classification basé sur la méthode de KNN**

Pour notre étude nous avons basé sur le logiciel Orange² qui implémente différents algorithmes de classification. La figure (1.4) illustre les résultats obtenus en classification basés sur l’algorithme de KNN, le choix de paramètre K est fixé à 10.

Tchebichef	Krawtchouk	Hahn
97.14	96.42	96.30

FIGURE 3.6 – Taux d’apprentissage pour chaque moment

- **Résultat obtenu en classification basé sur la méthode de rétropropagation -RNA-**

Nous présentons les résultats obtenus à partir d’algorithme de rétropropagation pour les différents moments, avec un choix de 10 et 20 couches cachées.

Tchebichef	Krawtchouk	Hahn
92.95	93.75	92.01

FIGURE 3.7 – Taux d’apprentissage d’algorithme de rétropropagation, 10 couches cachées

Tchebichef	Krawtchouk	Hahn
95.61	96.06	93.75

FIGURE 3.8 – Taux d’apprentissage d’algorithme de rétropropagation, 20 couches cachées

3.3.3 Reconnaissance de visage - BD YALE FACE -

La base de données YALE FACE [21], contient 165 images GIF de 15 sujets (subject01, subject02, etc), Il y en a onze images par sujet, chacune de ces onze images appartient à l’une des expressions du visage ou des configurations suivantes : centre-légers, heureux, gauche-lumière, avec et pas de lunettes, normal, non-lumière, triste, endormi, surpris et clin d’œil.

Dans notre étude nous avons opté pour le choix de six expressions par sujet comme donnée d’apprentissage et les cinq expressions qui reste pour la phase de test, les types choisis au niveau d’apprentissage sont : normal, avec et pas de lunettes, surpris, centre-légers et gauche-lumière. Les types restants sont admis à la phase de test, les figures (3.9, 3.10) au-dessous représentent successivement des exemples d’apprentissages et de tests.

Le test de validité sera appliqué juste pour les moments de Krawtchouk et de Tchebichef, les moments de Hahn seront éliminés, car ils ne supportent pas le calcul des moments pour des images de taille qui dépassent 64×64 d’une façon directe. Pour la taille des images nous avons

2. Logiciel Open source de visualisation des données et d’analyse pour les débutants et les experts. Il offre L’exploration de données grâce à une programmation visuelle. Pour plus d’information, visitez le site <http://orange.biolab.si>

réalisé un calcul d'échelle pour rendre les images dans la formes carrées de taille 128×128 sachant que leur taille originale est 240×320 .

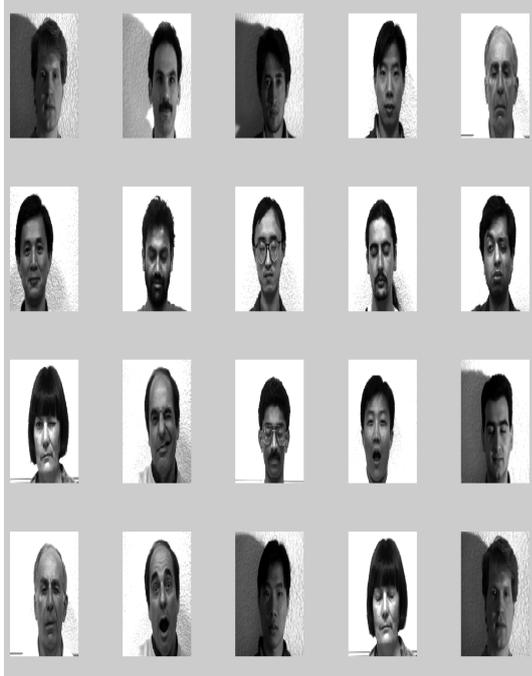


FIGURE 3.9 – Exemple d'apprentissage

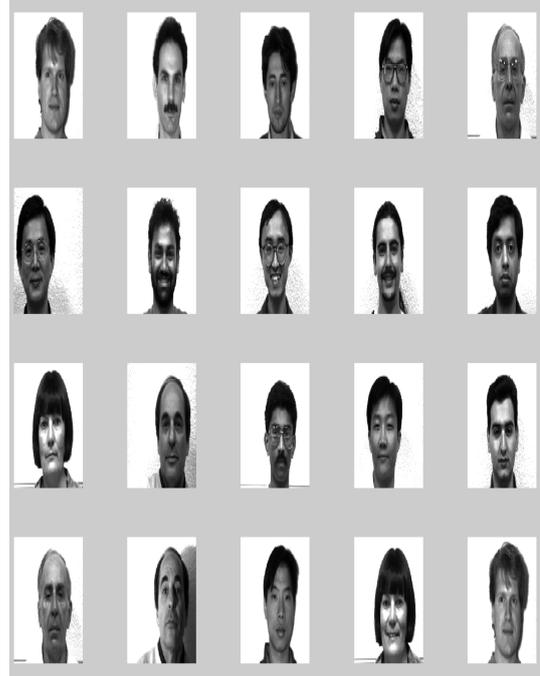


FIGURE 3.10 – Exemple de test

- **Résultat obtenu en classification basé sur la méthode de KNN**

La figure (3.11) exprime les résultats obtenus en classification basée sur l'algorithme de KNN, le choix de paramètre K est fixé à 20.

Tchebichef	Krawtchouk
72	80

FIGURE 3.11 – Taux d'apprentissage pour chaque moment

- **Résultat obtenu en classification basé sur la méthode de rétropropagation -RNA-**

Comme dans la partie précédente, nous avons opté pour le choix de 10 et 20 couches cachées successivement pour l'algorithme de rétropropagation, les figures(3.12, 3.13) au-dessous illustrent les résultats obtenus.

Tchebichef	Krawtchouk
82.67	81.33

FIGURE 3.12 – Taux d'apprentissage d'algorithme de rétropropagation, 10 couches cachées

Tchebichef	Krawtchouk
84	81.33

FIGURE 3.13 – Taux d'apprentissage d'algorithme de rétropropagation, 20 couches cachées

3.3.4 Analyse générale et discussion

D'après les deux algorithmes de classification proposés, nous pouvons conclure que les moments sont des attributs validés pour classier des objets au sens informatique. D'autre part, les moments de Tchebichef et Krawtchouk sont meilleurs par rapport au moment de Hahn au niveau de différents algorithmes de classification proposés, ce qui implique que les moments qui permettent de garantir un taux d'erreur minimal au niveau de reconstruction sont ceux qui sont adéquats pour la classification dans notre travail. Ainsi qu'au niveau de la BD-MNIST, les taux de classifications obtenus sont meilleurs par rapport aux taux obtenus en BD-YALE FACE.

3.4 Conclusion

Dans ce chapitre, nous avons présenté notre système de classification proposé dans notre travail, qui s'articule sur deux phases importantes : Phase d'extraction des caractéristiques et phase de classification, cette dernière est présentée par deux algorithmes qui s'intègrent dans l'ensemble des méthodes de classification supervisée (algorithme de KNN et rétropropagation). Au niveau des résultats obtenus on peut synthétiser que les moments qui ont présenté une bonne qualité au niveau de reconstruction sont les mêmes qui ont présenté des taux de classification élevés, donc nous concluons que les moments orthogonaux discrets de Tchebichef et Krawtchouk sont performants comme attributs de classification.

Chapitre 4

Application de reconstruction des Images 2-D en niveau de gris via les moments discrets

4.1 Introduction

Ce chapitre présente la partie pratique de projet, qui prend en charge l'implémentation d'une application permettant la reconstruction des images 2-D niveau de gris via les différents moments discrets étudiés dans notre recherche. Afin d'atteindre cet objectif nous présentons les différentes parties d'implémentation et les résultats obtenus au niveau de sortie de l'application.

4.2 Logiciel de Développement

MATLAB (MATrix LABoratory) est un logiciel basé sur le calcul matriciel. Son noyau est composé de bibliothèques écrites au début en Fortran puis en C++. Il permet :

- le calcul numérique.
- le tracé de graphiques.
- la programmation.
- l'interfaçage avec d'autres langages (Fortran, C++).
- le développement d'interface utilisateur (GUI).
- l'utilisation des boîtes à outils (Toolboxes) spécialisées.

MATLAB présente deux modes de fonctionnement :

1. Le mode interactif : exécution des instructions entrées par l'utilisateur dans la fenêtre de commande de MATLAB (le Workspace).
2. Le mode exécutif : exécution de "programmes" ou "script" qui sont des fichiers regroupant une suite d'instructions (les fichiers M).

4.3 Réalisation et constraints

Le but admis pour notre application est de pouvoir atteindre les différents types de reconstruction admis par les moments discrets étudiés dans notre recherche, donc on peut formuler ce besoin par les contraintes suivantes :

- Possibilité de reconstruction des images 2-D en niveau de gris ou binaires.
- Reconstruction des images bruitées.
- Traçage des courbes de PSNR et MSE dont le but est de faire des analyses si besoin.

L'application doit être lisible, bien claire et contenant des aides sur l'utilisation des différentes offres admises par elle.

4.4 Modélisation

Pour modéliser l'application souhaitée, nous proposons, le diagramme de cas d'utilisation et de séquences, dont l'objectif est la découverte de besoin de l'application.

- **Diagramme de cas d'utilisation**

Ce diagramme permet de représenter l'interaction entre l'acteur et le système dont lequel, nous présentons les fonctionnalités demandées par l'acteur qui seront construites dans le système. La figure (4.1) résume ces différentes données :

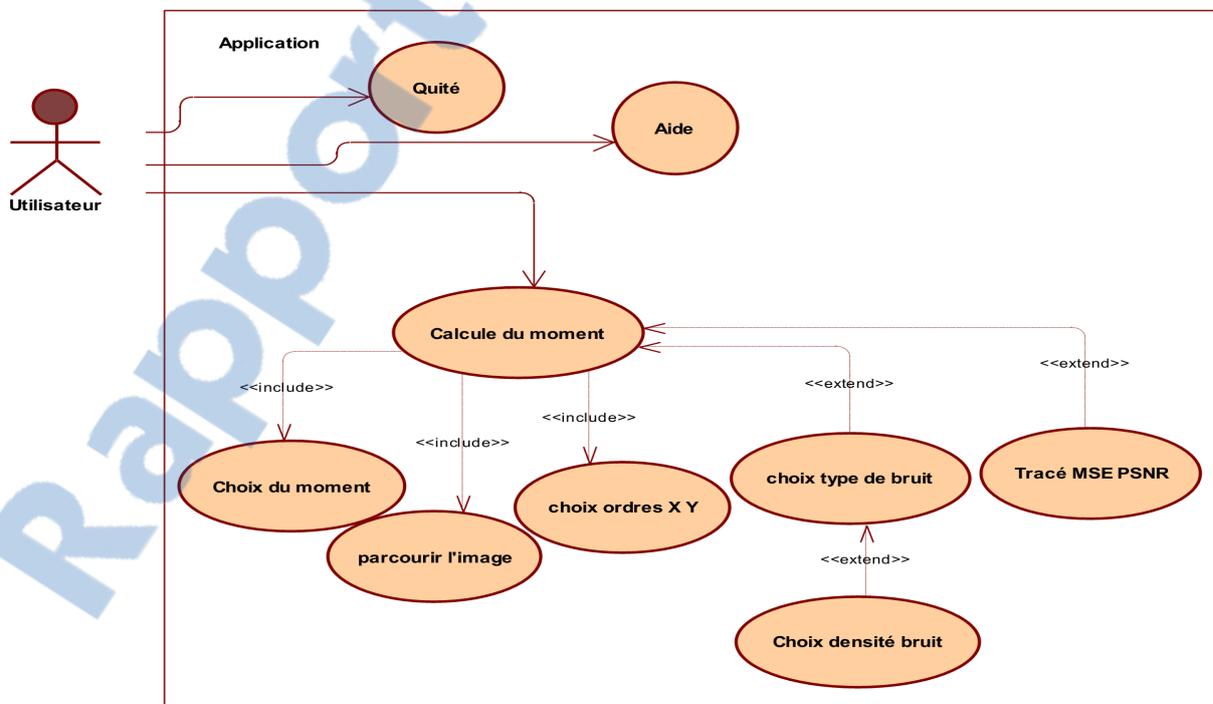


FIGURE 4.1 – diagramme de cas d'utilisation

- **Diagramme de séquence**

Ce diagramme permet de modéliser les échanges prévus par le système pendant le temps. Ce modèle inclut les activités qui seront réalisées au niveau d'une tâche exercée dans notre système.

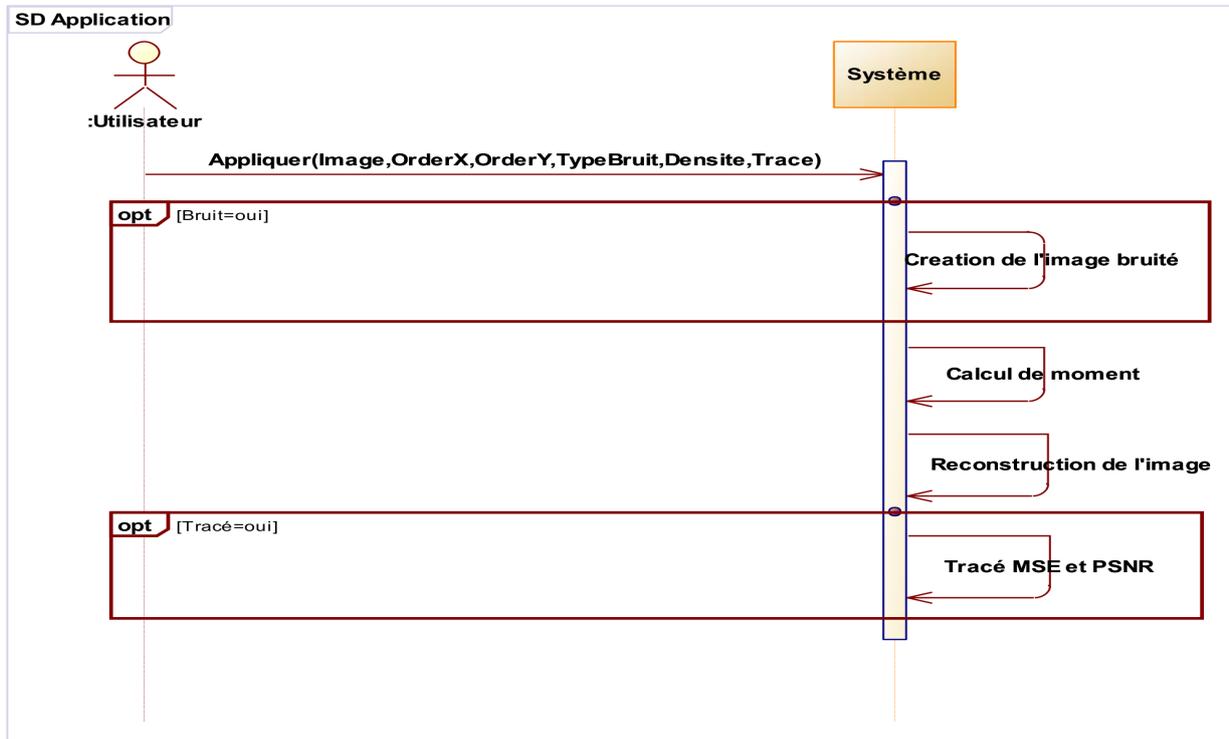


FIGURE 4.2 – Diagramme de séquence

4.5 Application Réalisée

Cette partie permet d'illustrer les différentes possibilités offertes par l'application, nous commençons par la présentation de l'accueil de notre application, qui est composé d'une image au fond, et une barre de menu comprenant trois menus, Fichier, Moment et Aide.

- Le menu Fichier contient trois menu Item, Ouvrir, Enregistrer sous et Fermer.
- Le menu Moment, contient quatre menus Item permettant l'ouverture d'une figure contenant les options attachées pour chaque type de moment.
- Menu Aide présente la procédure d'utilisation de l'application.

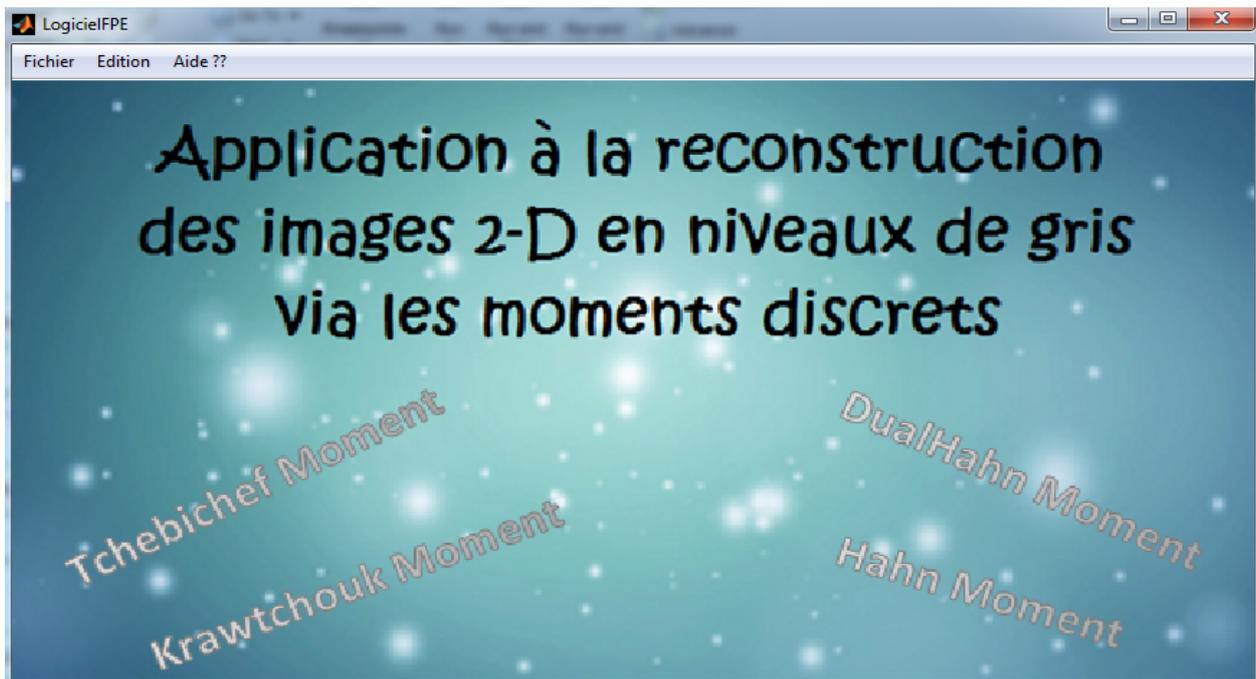


FIGURE 4.3 – accueil de l'application

Dans les articles suivants, nous présentons les différentes options de reconstruction admises pour chaque type de moment avec des exemples des images différentes.

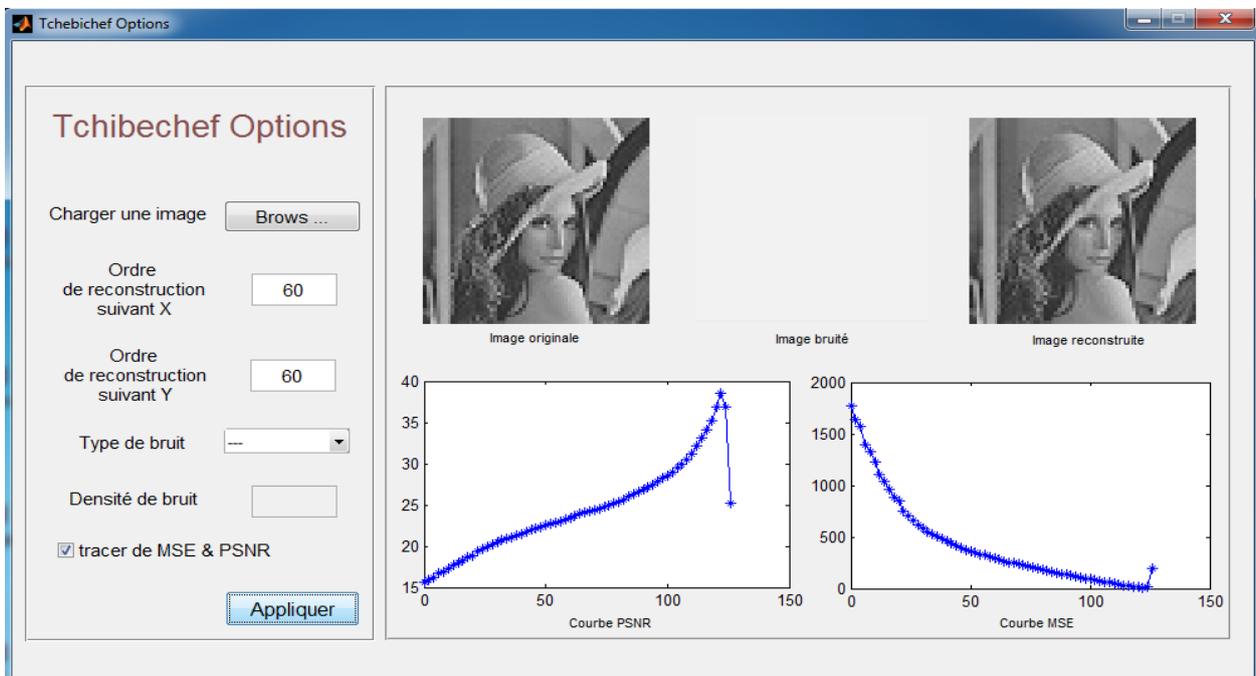


FIGURE 4.4 – Reconstruction de Lena via le moment de Tchebichef

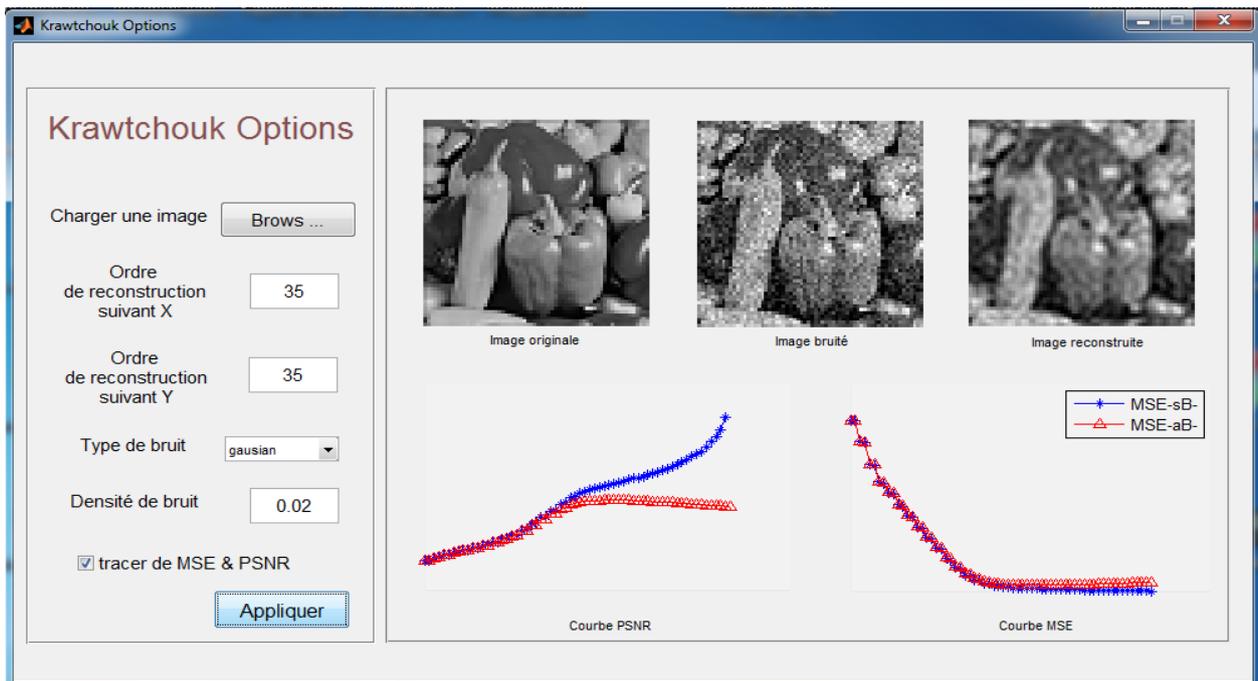


FIGURE 4.5 – reconstruction de Pepper via le moment de Krawtchouk, ajout de bruit gaussien

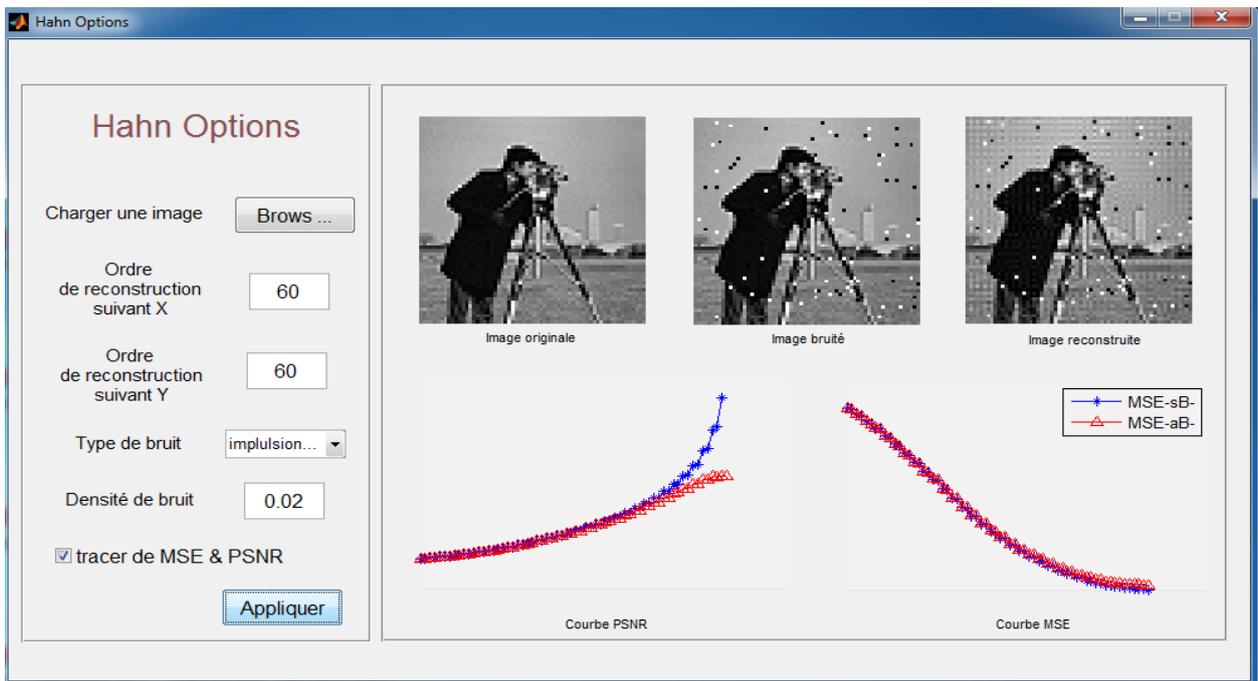


FIGURE 4.6 – reconstruction de Cameramman via le moment de Hahn, ajout de bruit impulsif

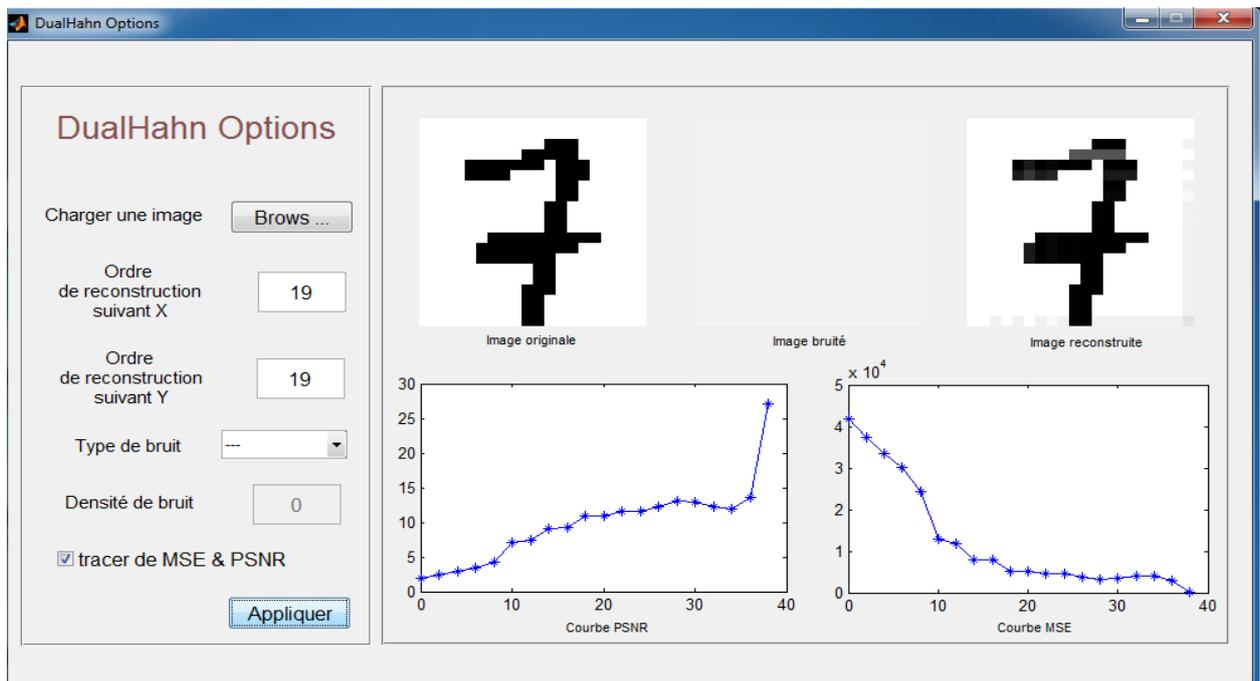


FIGURE 4.7 – reconstruction du chiffre 7 via le moment de DualHahn

4.6 Conclusion

Ce chapitre a présenté l'application réalisée au niveau de projet, cette application ne présente qu'un module, dont l'objectif est de le compléter dans des travaux futurs.

Conclusion Générale

Dans ce travail, nous nous sommes intéressés aux moments orthogonaux discrets et leurs applications en analyse d'image. Nous avons tout d'abord étudié, de manière approfondie, la théorie des moments, en réservant une attention particulière aux moments Tchebichef, Krawtchouk, Hahn et DualHahn.

En se basant sur cette étude, nous avons établi une comparaison en termes de pouvoir de reconstruction, des images binaires et niveau de gris, entre les différents types de moments orthogonaux discrets. Les résultats expérimentaux ont montré que les moments de Tchebichef et Krawtchouk donnent une meilleur qualité de reconstruction en terme de PSNR par rapport aux moments de Hahn et DualHahn.

Ensuite, nous avons proposé un système de reconnaissance d'image en utilisant les moments orthogonaux discrets de Tchebichef, Krawtchouk et Hahn comme descripteurs de forme. Notre système de classification proposé est constitué de deux étapes principales :

- *Extraction des attributs moments* : le vecteur attribut moment est obtenu en trois phases essentielles. Premièrement, On calcul les moments, après, l'ordre optimal des moments calculé et déterminé en utilisant le critère MSE. Enfin la phase d'extraction de vecteur attribut optimal pour chaque image de la base de données.
- *Classification* : Lors de cette étape on a utilisé deux algorithmes de classification à savoir ; le KNN (K-plus proches voisins) et ANN (Réseaux de neurones).

Pour avoir une évaluation pertinente de notre système de reconnaissance, nous l'avons testé sur deux bases de données hétérogènes, MNIST est utilisée a la reconnaissance des chiffres manuscrits et YALE FACE qui présente l'exemple de la reconnaissance de visage.

D'après les résultats obtenus au niveau de classification, on peut déduire que les moments qui ont un pouvoir de reconstruction élevé, donnent des taux de reconnaissance très important. Sachant qu'au niveau d'extraction de données caractéristiques aucun prétraitement n'est nécessaire, ce qui montre la robustesse des moments orthogonaux discrets comme attributs de classification.

Comme perspective, nous allons étudier d'une manière plus profonde la détermination du vecteur moment optimal en utilisant la théorie de probabilité, de même nous continuons l'implémentation d'autres types de moments dont l'objectif d'avoir une étude plus globale de tous les types des moments orthogonaux discrets qui existent dans la littérature. Finalement, nous continuons le développement de l'application déjà présentée, par l'ajout des méthodes de classification choisi au cours de notre travail.

Bibliographie

- [1] IRE TRANSACTIONS ON INFORMATION THEORY "Visual Pattern Recognition by Moment Invariants" MING-KUEI HU, vol. 49, pp. 830, January, 1961.
- [2] Computation Strategies of Orthogonal Image moments : A comparative study, G.A Papakostas, D.E Koulouriotis, E.G Karakasis, Applied Mathematics and Computation 216 (2010) 1-17.
- [3] Image Analysis by Tchebichef Moments, R.Mukundan Senior Member IEEE,S.H Ong, P.A Lee, Ong, IEEE TRANSACTIONS ON IMAGE PROCESSING, vol 10, NO 9, september 2001.
- [4] Image Analysis by Krawtchouk Moments, Pew-Thian Yap, Raveendran Paramesra, Senior Member IEEE, and Seng-Huat Ong, IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 12, NO. 11, NOVEMBER 2003
- [5] M. Krawtchouk, On interpolation by means of orthogonal polynomials, Memoirs Agricultural Inst. Kyiv 4 (1929) 21–28.
- [6] Image reconstruction using various discrete orthogonal polynomials in comparison with DCT, K.W. See , K.S. Loke, P.A. Lee, K.F. Loe, Applied Mathematics and Computation 193 (2007) 346–359.
- [7] J. Zhou, H. Shu, H. Zhu, C. Toumoulin, L. Luo, Image analysis by discrete orthogonal Hahn moments, in : ICIAR 2005, LNCS, vol. 3656, 2005, pp. 524–531.
- [8] Image analysis by discrete orthogonal dual Hahn moments by Hongqing Zhu, Huazhong Shu, Jian Zhou, Limin Luo et J.L. Coatrieux.Pattern Recognition Letters 28 (2007) 1688–1704.
- [9] Algorithme des K plus proches voisins pondérés (WKNN) et Application en diagnostic, Eve MATHIEU-DUPAS, Manuscrit auteur, publié dans 42èmes Journées de Statistique (2010).
- [10] Le perceptron multicouche et son algorithme de rétropropagation des erreurs, Marc Parizeau, Département de génie électrique et de génie informatique Université Laval, 10 septembre 2004.
- [11] Recognitive Aspects of Moment Invariants,YASER S. ABU-MOSTAFA AND DEMETRI PSALTIS,IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. PAMI-6, NO. 6, NOVEMBER 1984.

- [12] C. -H Teh, R.T. Chin, On image analysis by the method of moments, *IEEE Trans. Pattern Anal. Mach. Intel.* 10 (1988) 496–513.
- [13] R. Mukundan, K.R. Ramakrishnan, *Moment Functions in Image Analysis—Theory and Applications*, World Scientific, Singapore, 1998.
- [14] M.R. Teague, Image analysis via the general theory of moments, *J. Opt. Soc. Amer.* 70 (1980) 920–930.
- [15] Lo, C. H., Don, H. S. : 3D moment forms : Their construction and application to object identification and positioning, *IEEE Trans. Pattern Anal. Mach. Intell.* 11 (1989), 1053–1064.
- [16] Flusser, J., Suk, T. : Pattern Recognition by affine moment invariants, *Pattern Recognition*, 26 (1993), 167–174.
- [17] Liao, S. X., Pawlak, M. : On image analysis by moments, *IEEE Trans. Pattern Anal. Mach. Intell.* 18 (1996), 254–266.
- [18] Luo, L. M., Hamitouche, C., Dilenseger, J. L., Coatrieux, J. L. : A moment-based three-dimensional edge operator, *IEEE Trans. Biomed. Eng.* 40 (1993), 693–703.
- [19] Luo, L. M., Xie, X. H., Bao, X. D. : A modified moment-based edge operator for rectangular pixel image, *IEEE Trans. Circuits Systems Video Technol.* 4 (1994), 552–554.
- [20] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, “Gradient-Based Learning Applied to Document Recognition,” *Proceedings of the IEEE.*, vol. 86, no. 11, 1998, pp. 2278-2324.
- [21] Yale Face Database, <http://vision.ucsd.edu>