

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mohamed Khider Biskra
Faculté des Sciences Exactes et des Sciences de la Nature et de la Vie
Département d'Informatique

N ° d'ordre :

Série :



Mémoire

Présenté en vue de l'obtention du diplôme de

Magister en Informatique

Option : Data Mining et Multimédia

Thème

Approche Quantique pour l'Appariement de Formes

Par

Mohamed Khalil MEZGHICHE

Soutenu publiquement le : 04 / 02 / 2015

Devant le jury composé de :

Mohamed Chaouki BABAHENINI	MCA	Université de Biskra	Président
Kamal Eddine MELKEMI	MCA	Université de Biskra	Rapporteur
Foudil CHERIF	MCA	Université de Biskra	Examineur
Ammar LAHLOUHI	MCA	Université de Batna	Examineur

Je dédie ce travail à :

Mes chers parents ...

Mes frères et ma sœur ...

Toute la famille ...

Et tous mes amis ...

Remerciements

Tout d'abord, je remercie Allah, le tout puissant, qui m'a donné la force, la patience et la volonté pour accomplir ce modeste travail.

Je tiens à remercier le docteur Kamal Eddine Melkemi pour son encadrement, sa disponibilité, son suivi, ses conseils précieux et son encouragement.

Mes sincères remerciements à Dr Mohamed Chaouki Babahenini qui m'a fait l'honneur de présider le jury de soutenance. J'adresse également tous mes remerciements à Dr Foudil Cherif et Dr Ammar Lahlouhi pour avoir accepté d'évaluer mon travail et de participer au jury de soutenance.

Je remercie également le Pr Sebti Foufou pour ses contributions et la correction de l'article.

Je tiens à remercier toute personne ayant contribué de près ou de loin à la réalisation de ce travail.

Résumé

L'appariement de formes est un sujet important dans la vision par ordinateur, il étudie la mesure de similarité entre les formes, il représente un composant essentiel dans la recherche de formes, la reconnaissance, la classification et le recalage. Dans ce travail, nous avons essayé d'aborder l'appariement de formes comme un problème d'optimisation, à l'aide des algorithmes évolutionnaires quantiques qui ont démontré le pouvoir d'être une amélioration majeure par rapport aux algorithmes évolutionnaires classiques. Nous proposons de combiner le descripteur shape context avec les algorithmes génétiques quantiques afin de définir une nouvelle approche d'appariement et de recherche de formes. L'appariement de formes avec shape context est basé sur l'idée de trouver la meilleure correspondance entre deux ensembles de points échantillonnés à partir des deux formes. Notre approche proposée utilise les algorithmes génétiques quantiques pour trouver la meilleure configuration de points afin d'obtenir le meilleur appariement possible entre les deux formes. Notre algorithme proposé (Quantum Shape Context) est également utilisé pour résoudre une des faiblesses de shape context, sa faible invariance à la rotation et le retournement, où nous utilisons les algorithmes génétiques quantiques pour estimer la meilleure orientation de la forme cible qui permet d'obtenir le meilleur appariement pour les formes avec des rotations et retournements. Les résultats expérimentaux ont montré la supériorité de notre algorithme proposé dans les tâches d'appariement et de recherche de formes par rapport au shape context et d'autres méthodes d'appariement de formes contre divers types de tests.

Mots clés : appariement de formes, recherche de formes, shape context, descripteur de forme, mesure de similarité, algorithme génétique quantique, l'informatique quantique, algorithmes évolutionnaires quantiques.

Abstract

Shape matching is an important subject in computer vision, it studies the similarity measure between shapes, it represent an essential component in shape retrieval, recognition, classification and registration. In this work, we tried to solve the shape matching as an optimization problem, using the quantum evolutionary algorithms which demonstrated to be a major improvement over the classical evolutionary algorithms. We propose to combine the well known shape context descriptor with quantum genetic algorithms in order to define a new shape matching and retrieval method. The shape context matching method is based on the idea of finding the best correspondence between two points sets sampled from the two shapes. Our proposed method uses the quantum genetic algorithms to find the best configuration of sample points in order to achieve the best possible matching between the two shapes. Our proposed Quantum Shape Context algorithm is also used to solve one of the weaknesses of shape context, its weak rotation and flipping invariance, where we use the quantum genetic algorithms to estimate the best orientation of the target shape that allows obtaining the best matching result for rotated and flipped shapes. The experimental results shows the superiority of our proposed algorithm in shape matching and shape retrieval tasks comparing to shape context and some other shape matching methods against various forms of tests.

Keywords : shape matching, shape retrieval, shape context, shape descriptor, similarity measure, quantum genetic algorithm, quantum computing, quantum evolutionary algorithms.

ملخص

تعتبر مطابقة الأشكال إحدى الموضوعات الهامة في الرؤية الحاسوبية، حيث تدرس قياس التشابه بين الأشكال، لذلك فإنها تمثل عنصراً أساسياً في استرجاع الأشكال، التعرف، التصنيف والمواءمة. في هذا العمل، حاولنا حل مطابقة الأشكال من خلال اعتبارها مشكلة استمثال باستخدام الخوارزميات التطورية الكمية، والتي أثبتت أنها تحسّن كبير للخوارزميات التطورية الكلاسيكية. نقترح الجمع بين shape context والخوارزميات الجينية الكمية من أجل استحداث طريقة مطابقة واسترجاع جديدة. وتستند طريقة shape context لمطابقة الأشكال على فكرة إيجاد أفضل توافق بين مجموعتين من النقاط مستخرجة من الشكلين. الطريقة المقترحة تستخدم الخوارزميات الجينية الكمية للعثور على أفضل مجموعة من النقاط المستخرجة من كل شكل، من أجل تحقيق أفضل مطابقة ممكنة بين الشكلين. نستخدم الخوارزمية المقترحة أيضاً من أجل حل واحدة من نقاط الضعف في طريقة shape context، وهي ضعفها أمام الأشكال التي تم تدويرها أو قلبها، حيث نستخدم الخوارزميات الجينية الكمية لتقدير أفضل توجه للشكل الهدف الذي يسمح بالحصول على أفضل مطابقة ممكنة للأشكال التي تم تدويرها أو قلبها. وأظهرت النتائج التجريبية تفوق الخوارزمية المقترحة في عمليات المطابقة واسترجاع الأشكال مقارنة بالطرق السابقة ضد أنواع مختلفة من الاختبارات.

كلمات مفتاحية : مطابقة الأشكال، الاعلام الالي الكمي، الخوارزميات التطورية الكمية، مقياس التشابه، واصف الشكل، الخوارزميات التطورية الكمية، الخوارزميات الجينية الكمية، استرجاع الأشكال.

Table des matières

Dédicace	i
Remerciements	ii
Résumé	iii
Abstract	iv
Résumé en arabe	v
Table des matières	viii
Liste des figures	x
Liste des tableaux	xi
Introduction générale	1
1 Informatique quantique	4
1.1 Introduction	5
1.2 Mécanique quantique	5
1.2.1 Espace de Hilbert	6
1.2.2 Notation de Dirac	6
1.2.3 Postulats de la mécanique quantique	7
1.3 Notions de l'informatique quantique	8
1.3.1 Bit quantique	8
1.3.2 Réalisation physique d'un bit quantique	9
1.3.3 Registre quantique	11
1.3.4 Principes de l'informatique quantique	12
1.3.5 Mesure quantique	13
1.3.6 Calcul quantique	14
1.3.7 Portes quantiques	14
1.3.8 Circuits quantiques	17
1.3.9 Cryptographie quantique	18
1.3.10 Correction d'erreurs quantiques	20
1.3.11 Téléportation quantique	21

1.3.12	Critères de la réalisation d'un ordinateur Quantique	22
1.3.13	Algorithmes quantiques	22
1.3.14	Ordinateurs quantiques	25
1.4	Conclusion	27
2	Algorithmes évolutionnaires quantiques	28
2.1	Introduction	29
2.2	Algorithmes évolutionnaires	29
2.2.1	Classes des algorithmes évolutionnaires	30
2.2.2	Principes des algorithmes évolutionnaires	31
2.3	Principes des algorithmes évolutionnaires quantiques	34
2.3.1	Représentation quantique dans les AEQs	35
2.3.2	Observation dans les AEQs	36
2.3.3	Opérations de mise à jour	37
2.3.4	Structure générale des AEQs	39
2.3.5	Types des AEQs	41
2.3.6	Caractéristiques des AEQs	42
2.3.7	Renforcement et Diversification dans les AEQs	43
2.3.8	Avantages des AEQs	44
2.4	Conclusion	44
3	Decripteurs et mesures de similarité	45
3.1	Introduction	47
3.2	Quelques définitions	47
3.3	Paramètres de forme	48
3.3.1	Centre de gravité	48
3.3.2	Axe de moindre inertie	48
3.3.3	Énergie de pliage	49
3.3.4	Excentricité	49
3.3.5	Rapport de circularité	49
3.3.6	Rectangularité	50
3.3.7	Convexité	50
3.3.8	Solidité	50
3.3.9	Nombre d'Euler	50
3.3.10	Profils	51
3.3.11	Rapport de la surface du trou	51
3.3.12	Invariance aux transformations géométriques	51
3.4	Descripteurs de formes	52
3.4.1	Descripteurs basés région	55
3.4.2	Descripteurs basés contour	57

3.4.3	Ensemble fini de points	59
3.4.4	L'invariance et la résistance des descripteurs de forme	60
3.5	Mesures de similarité	60
3.5.1	Distance métrique	61
3.5.2	Distance non métrique	64
3.6	Transformations	66
3.6.1	Translation	66
3.6.2	Transformation rigide	66
3.6.3	Similitude	66
3.6.4	Transformation affine	67
3.6.5	Transformation projective	67
3.7	Conclusion	67
4	Approche évolutionnaire quantique pour l'appariement de formes	69
4.1	Introduction	70
4.2	Principes de l'algorithme évolutionnaire quantique pour l'appariement de formes	70
4.2.1	Descripteur de forme shape context	70
4.2.2	Représentation quantique	73
4.2.3	Fonction objectif	75
4.2.4	Stratégie de recherche	75
4.2.5	Algorithme QSC proposé	77
4.3	Conclusion	80
5	Evaluation de l'approche proposée	82
5.1	Introduction	83
5.2	Expérimentations et résultats	83
5.2.1	Matériels et logiciels utilisés	83
5.2.2	Protocole d'évaluation	84
5.2.3	Etude de l'effet de θ sur la convergence de l'algorithme	84
5.2.4	Etude de l'effet de nombre d'individus sur la convergence de l'algorithme	86
5.2.5	Recherche d'images	88
5.2.6	Temps de calcul	100
5.3	Conclusion	102
	Conclusion générale	103
	Bibliographie	105

Table des figures

1.1	La sphère de Bloch	8
1.2	Atome à deux niveaux	10
1.3	Décomposition de la polarisation par une lame biréfringente	11
1.4	L'observation (mesure) quantique	13
1.5	La porte quantique U-Contrôlé $\Lambda^k U$	15
1.6	La porte quantique Controlled-Not	16
1.7	La porte quantique Toffoli	16
1.8	La porte quantique racine-carré-du-not	17
1.9	Le circuit quantique $\Lambda^5 U$	18
1.10	Un code correcteur à trois bits	21
2.1	Les classes des algorithmes évolutionnaires	30
2.2	Le cycle d'un algorithme évolutionnaire	32
2.3	Le croisement	34
2.4	La mutation	34
2.5	La porte de rotation quantique	38
2.6	Le croisement quantique	38
2.7	La mutation quantique	39
2.8	La permutation quantique	39
2.9	Les types des AEQs	41
3.1	La méthode des axes principaux	49
3.2	Une enveloppe convexe	50
3.3	Les techniques de représentation de forme	54
3.4	La classification des descripteurs de forme	54
3.5	Une matrice de forme	56
3.6	La construction du squelette	57
3.7	L'effet d'augmentation de la valeur de σ	59
3.8	La distance intérieure entre deux points	60
3.9	La représentation des distances Manhattan, Euclidienne et Cosinus	63
3.10	La distance de Hausdorff	64
3.11	La perception humaine de similarité	65
4.1	Le descripteur shape context	71
4.2	La transformation thin-plate-splines (TPS)	72

4.3	La sélection des points	73
4.4	Contrôler l'orientation de la forme	74
4.5	La matrice de coût	76
4.6	L'opérateur de rotation quantique	77
4.7	La sélection de la meilleure solution globale	80
4.8	L'algorithme QSC	80
5.1	Image bat-1 à gauche et bat-2 à droite	85
5.2	Les résultats d'exécution avec $\theta= 0.04\pi$	85
5.3	Les résultats d'exécution avec $\theta= 0.02\pi$	85
5.4	Les résultats d'exécution avec $\theta= 0.01\pi$	86
5.5	Les résultats d'exécution avec 2 individus	87
5.6	Les résultats d'exécution avec 4 individus	87
5.7	Les résultats d'exécution avec 8 individus	87
5.8	La base d'images MPEG-7 CE-shape-1	88
5.9	Le graphe de rappel-précision sur la base MPEG-7	91
5.10	La base d'images Kimia 25	92
5.11	La base d'images Kimia 99	93
5.12	Le graphe rappel-précision sur la base Kimia 99	96
5.13	La base COIL-100	97
5.14	Des formes obtenues à partir des images couleur	97
5.15	Le graphe rappel-précision sur la base COIL-100	100
5.16	Une comparaison des temps de calculs pour SC et QSC	101

Liste des tableaux

1.1	Comparaison entre le bit classique et le bit quantique	9
1.2	Comparaison entre le nombre de qubits et le nombre équivalent des bits classiques	12
1.3	La composition d'un message secret	19
2.1	L'angle et la direction de rotation	38
3.1	L'invariance des paramètres de la forme	52
3.2	L'invariance et la résistance des descripteurs de forme	61
4.1	Le tableau de référence	77
5.1	Les résultats de l'étude de l'effet de θ	86
5.2	Les résultats de l'étude de l'effet de nombre d'individus	87
5.3	Les résultats de recherche de formes sur la base d'images MPEG-7	90
5.4	Les résultats de recherche de formes sur la base MPEG-7	91
5.5	Les résultats de recherche de formes sur la base d'images Kimia 25	92
5.6	Les résultats de recherche de formes sur la base d'images Kimia 99	95
5.7	Les résultats de recherche de formes sur la base Kimia 99	96
5.8	Les résultats de recherche de formes sur la base d'images COIL-100	99
5.9	Les résultats de recherche de formes sur la base COIL-100	100
5.10	les temps de calculs pour la recherche d'images sur la base Kimia 99	101

Introduction générale

L'information visuelle joue un rôle très important dans notre vie quotidienne, plusieurs domaines comme le journalisme, la publicité, l'architecture et la médecine utilisent des applications basées principalement sur l'information visuelle. L'exploitation de cette grande quantité d'images nécessite des mécanismes efficaces pour l'analyse, la classification et l'indexation. La vision par ordinateur est la branche en informatique qui prend en charge l'acquisition, le traitement et l'analyse de l'information visuelle. Le but essentiel des recherches dans ce domaine est le développement d'un système capable de reproduire les capacités de la vision humaine, de percevoir et de comprendre une image. L'appariement de formes représente un ingrédient essentiel dans l'interprétation de l'information visuelle, dont l'objectif est de mesurer la similarité entre les formes.

L'appariement de formes est un problème central dans la vision par ordinateur, la reconnaissance d'objets et la robotique. Il étudie la transformation et la mesure de similarité entre deux formes [Vel01]. La représentation de formes et la mesure de similarité entre les formes représentent les éléments les plus importants de l'appariement de formes. Une mesure de similarité est une fonction de distance qui associe une valeur numérique à deux descripteurs de forme, cette distance est faible si les deux formes sont similaires, et plus importante si ces formes sont dissimilaires.

Dans la littérature, plusieurs méthodes d'appariement de formes sont décrites, parmi celles-ci, nous avons les méthodes basées sur les descripteurs de forme globale comme, CSS (Curvature Scale Space) [MAK96], Transformée de Hough [Bal81], le descripteur de Fourier [Lon98], la distance de Hausdorff [Ruc96], ces méthodes ne sont pas robustes à l'occultation et à l'articulation, et l'appariement partiel n'est pas possible. Les techniques d'appariement de formes doivent être robustes aux transformations géométriques (translation, rotation, mise à l'échelle, etc.) et les déformations non-linéaires (bruit, articulation, occultation). Par rapport aux transformations géométriques, les déformations non-linéaires sont beaucoup plus difficiles. Les méthodes basées sur l'ensemble fini de points sont plus robustes, Shape Context (SC) [BMP00] est parmi ces méthodes, c'est un descripteur riche et robuste face à la plupart des transformations géométriques. Inner-Distance Shape Context (IDSC) [LJ07] est une amélioration de shape context afin d'être plus robuste aux articulations. D'autres descripteurs ont été proposés pour améliorer la performance de shape context, comme Skeletal Contexts [XHS08], et Aspect Shape

Context [LYL10].

Dans ce mémoire, nous proposons une nouvelle approche d'appariement de formes basée sur les concepts de l'informatique quantique et le descripteur shape context dénotée par (QSC)[MMF14]. L'informatique quantique est basée sur les principes de la mécanique quantique comme la superposition d'états et la mesure probabiliste. La différence essentielle entre l'informatique classique et l'informatique quantique repose dans la représentation de bit, qui peut avoir une des deux valeurs 0 ou 1 en informatique classique, alors qu'un bit quantique peut avoir la valeur 0 ou 1, mais il peut être aussi dans une superposition des deux états en même temps. En conséquence, un registre quantique de n qubits peut contenir 2^n valeurs distinctes en superposition quantique, donc il serait capable de calculer à la fois les 2^n résultats correspondants à toutes les valeurs d'entrées en une seule opération, ce qui représente des capacités de traitement et de stockage exponentiel.

Les concepts de l'informatique quantique purs sont encore loin d'être appliqués tant que des vrais ordinateurs quantiques ne sont pas encore réalisés. Dans notre travail, nous allons nous baser sur une approche hybride, qui consiste à combiner les concepts inspirés de l'informatique quantique avec les algorithmes évolutionnaires, nous appelons cette technique les algorithmes évolutionnaires inspirés du quantique (AEQ).

Dans ce travail, nous avons opté d'aborder le problème de l'appariement de formes comme un problème d'optimisation par les algorithmes évolutionnaires quantiques. Notre approche combine l'appariement de forme avec le descripteur shape context et les AEQs pour obtenir les meilleurs résultats d'appariement de formes possibles. L'appariement avec shape context repose sur l'idée de trouver la meilleure correspondance entre les points extraits du contour des deux formes, alors que notre algorithme évolutionnaire quantique essaie plusieurs configurations de points afin de trouver l'ensemble le plus adéquat où chaque configuration de points représente une version différente de shape context.

Notre approche quantique d'appariement de formes est fondée sur une représentation quantique de l'espace de recherche à l'aide d'un ensemble d'individus quantiques, chaque individu représente une solution possible, qui peut être vue comme une configuration de points capable de construire le descripteur shape context. La population quantique sera évaluée ensuite selon une fonction objectif (fitness) qui consiste au coût d'appariement minimum entre les deux formes. L'exploration de l'espace de recherche est réalisée par un opérateur de mise à jour quantique qui permet de guider la population vers les bonnes solutions jusqu'à la convergence de l'algorithme.

Les expérimentations avec notre approche proposée sont réalisées sur des bases d'images connues comme : MPEG-7 CE-shape-1 [LLE00], KIMIA25 [SCTK98], KIMIA99 [SKK04] et COIL-100 [NNM96]. Les résultats d'appariement et la recherche d'images obtenus sont

très prometteurs vis-à-vis de SC et de IDSC et grâce aux concepts de la théorie quantique, la convergence peut être atteinte avec moins d'itérations et en utilisant une petite population composée de quelques individus.

Le travail présenté dans ce mémoire est organisé en cinq chapitres :

Dans le premier chapitre, nous introduisons l'informatique quantique, nous présentons ses débuts à partir de la mécanique quantique et quelques concepts principaux. Ensuite, nous présentons le calcul quantique, les circuits quantiques, les algorithmes quantiques et deux exemples des algorithmes quantiques célèbres (Shor [Sho94] et Grover [Gro96]). Enfin, nous exposons les différents axes de recherche active dans la réalisation d'un ordinateur quantique.

En deuxième chapitre, nous présentons les divers types d'algorithmes évolutionnaires et le principe de fonctionnement de ces algorithmes. Ensuite, nous présentons les algorithmes évolutionnaires inspirés du quantique, la structure générale de ces algorithmes, la manière de représentation des individus quantiques ainsi que les opérateurs de mise à jour utilisés. Enfin, nous citons les différents types des algorithmes évolutionnaires quantiques et leurs avantages par rapport aux algorithmes évolutionnaires classiques.

Le troisième chapitre est consacré à la présentation de l'appariement de formes, sa définition, son objectif et son utilité dans un système de vision par ordinateur. Ensuite, nous présentons les différentes techniques de représentation de formes et les paramètres de formes, ainsi que les mesures de similarité utilisées. Enfin, nous citons quelques transformations géométriques les plus connues.

Notre approche évolutionnaire quantique pour l'appariement de formes est présentée dans le quatrième chapitre. Nous commençons par une introduction à l'appariement avec shape context. Ensuite, nous expliquons la représentation quantique qui nous permet d'utiliser ce descripteur pour notre approche ainsi que la stratégie de recherche appliquée. Enfin, nous exposons le fonctionnement détaillé de notre algorithme.

L'évaluation de la performance de notre approche proposée est exposée dans le cinquième chapitre où nous présentons quelques résultats expérimentaux obtenus par notre approche et des résultats comparables obtenus par d'autres méthodes d'appariement de formes.

Informatique quantique

Dans ce chapitre, nous présentons l'informatique quantique, en commençant par une introduction à la mécanique quantique. Nous exposons ensuite les concepts de base de l'informatique quantique et quelques exemples d'applications possibles. Enfin, nous clôturons le chapitre par une présentation des différents axes de recherche active dans la réalisation d'un ordinateur quantique.

1.1	Introduction	5
1.2	Mécanique quantique	5
1.2.1	Espace de Hilbert	6
1.2.2	Notation de Dirac	6
1.2.3	Postulats de la mécanique quantique	7
1.3	Notions de l'informatique quantique	8
1.3.1	Bit quantique	8
1.3.2	Réalisation physique d'un bit quantique	9
1.3.3	Registre quantique	11
1.3.4	Principes de l'informatique quantique	12
1.3.5	Mesure quantique	13
1.3.6	Calcul quantique	14
1.3.7	Portes quantiques	14
1.3.8	Circuits quantiques	17
1.3.9	Cryptographie quantique	18
1.3.10	Correction d'erreurs quantiques	20
1.3.11	Téléportation quantique	21
1.3.12	Critères de la réalisation d'un ordinateur Quantique	22
1.3.13	Algorithmes quantiques	22
1.3.14	Ordinateurs quantiques	25
1.4	Conclusion	27

1.1 Introduction

Selon la loi de Moore le cofondateur d'Intel [M⁺65], qui prédit le doublement du nombre des transistors dans une puce tous les deux ans, par conséquent avant l'année 2025 les transistors auront la taille d'un atome, Moore lui-même en 2007 a annoncé que la technologie actuelle basée sur les transistors approche de ses limites physiques, et ne serait plus valide dans dix à quinze ans.

La transition de la microélectronique aux nanotechnologies est devenue inévitable, d'ici les lois de la physique classique ne seront plus valables, et les lois de la physique quantique prennent place. La physique quantique nous permet d'entrer à un monde infiniment petit et étrange, le monde des photons et atomes, qui se comportent d'une manière contredit toutes les lois de la physique classique.

L'informatique quantique est fondée sur les lois de la mécanique quantique, bénéficier des concepts et principes puissants de cette théorie comme la superposition d'états, l'enchevêtrement et l'interférence. L'informatique quantique est encore à ses débuts, cependant elle présente un domaine de recherche très prometteur.

1.2 Mécanique quantique

La mécanique quantique est une des théories scientifiques les plus révolutionnaires du vingtième siècle, c'est la branche de la physique qui sert à décrire la structure et l'évolution dans le temps et l'espace des phénomènes physiques à l'échelle de l'atome. Les études sur le comportement des atomes et les photons ont mené à la découverte de cette théorie fascinante [Gro02].

Les travaux de Planck et Einstein sur les échanges d'énergie lumineuse [BM09] et les tentatives pour résoudre le problème de rayonnement du corps noir ont marqué le début de la mécanique quantique. Deux théories sont apparues, la mécanique ondulatoire de Broglie et Schrödinger et la mécanique matricielle de Heisenberg, Born et Jordan. Ces deux théories ont été unifiées par Dirac [Dir58] qui a défini une représentation mathématique pour la mécanique quantique basée sur un ensemble d'axiomes.

Les concepts de la mécanique quantique sont un peu étrange et difficile à comprendre comme la superposition quantique où une particule peut-être dans deux états en même temps, ou l'enchevêtrement des particules où deux particules dans deux endroits physiquement séparés sont dépendantes. Tous ces phénomènes sont inexplicables par les lois de la physique classique, ce qui nous donne une idée sur la puissance de la mécanique quantique.

1.2.1 Espace de Hilbert

Les différences, entre la physique classique représentée par l'espace euclidien et la physique quantique, mènent à l'introduction d'un nouveau formalisme mathématique qui peut prendre en charge les concepts de la mécanique quantique, ce formalisme repose sur les notions de l'espace de Hilbert.

L'espace de Hilbert est la version de dimension infinie de l'espace euclidien ou hermitien, un espace de Hilbert est un espace vectoriel pourvu d'un produit scalaire qui donne la possibilité de mesurer des longueurs et des angles. L'origine des espaces de Hilbert est la théorie de développement des fonctions arbitraires en séries des fonctions orthogonales. Les espaces de Hilbert fournissent la base mathématique pour la mécanique quantique [Che01].

L'état d'un système physique est une description globale de ce système, en mécanique quantique l'état est représenté par un vecteur normé dans l'espace de Hilbert H [Sch99].

1.2.2 Notation de Dirac

La notation bra-ket introduite par Dirac [Dir58] utilisée par les physiciens en mécanique quantique afin de faciliter l'écriture des équations, et la représentation du vecteur d'état quantique, cette notation appelée bra-ket qui provient du mot anglais bracket qui signifie crochet, Bra $\langle |$ et Ket $| \rangle$.

Par exemple, on prend les valeurs de l'état de bit $\{0, 1\}$, la base orthonormé $\{|0\rangle, |1\rangle\}$ peut être exprimée en tant que $\left\{\begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}\right\}$. Notons bien que :

- $|x\rangle$ est un vecteur utilisé pour décrire un état quantique, alors que $\langle x|$ est le vecteur transposé de $|x\rangle$.
- La composition de $|x\rangle$ avec $|y\rangle$ en $\langle x| |y\rangle$ ou $\langle x | y \rangle$, dénote le produit interne de $|x\rangle$, $|y\rangle$.
- La combinaison de $|x\rangle$ avec $\langle y|$ dénote le produit externe des deux vecteurs $|x\rangle$, $\langle y|$.

Par exemple le produit de deux vecteurs $|0\rangle \langle 1|$:

$$|0\rangle \langle 1| = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \quad (1.1)$$

1.2.3 Postulats de la mécanique quantique

Mathématiquement la mécanique quantique est très bien définie, sur la base de la notation bra-ket de Dirac [Dir58], qui permet la représentation des opérations sur les espaces de Hilbert [Pou02]. Cette formulation mathématique est le résultat des travaux de Dirac et John von Neumann [VH⁺58]. L'état d'un système quantique peut être décrit mathématiquement par les postulats de la mécanique quantique.

Postulat 1 : (l'état quantique) à un instant t_0 , l'état d'un système physique est défini par un vecteur d'état $|\psi(t_0)\rangle$ appartenant à l'espace des états H de Hilbert [BF92].

Postulat 2 : (description d'une grandeur physique) toute grandeur physique mesurable O est décrite par un opérateur \hat{O} agissant dans H , cet opérateur est appelé observable.

Postulat 3 : (mesures des grandeurs physiques) la mesure d'une grandeur physique O ne peut donner comme résultat qu'une des valeurs propres de l'observable \hat{O} correspondante.

Postulat 4 : (décomposition spectrale) lorsqu'on mesure une grandeur physique O sur un système dans l'état $|\psi(t_0)\rangle$ normé, la probabilité $Pr(a_n)$ d'obtenir comme résultat la valeur propre a_n de l'observable \hat{O} correspondante est :

$$Pr(a_n) = \sum_{j=0}^{g_n} |\langle \phi_n^j | \psi \rangle|^2$$

Où g_n est la dégénérescence de la valeur propre a_n et les $|\phi_n^j\rangle$ sont les vecteurs propres normés de \hat{O} associés à la valeur propre a_n .

Postulat 5 : (réduction du paquet d'ondes) si la mesure de la grandeur physique O sur le système dans l'état $|\psi\rangle$ donne le résultat a_n , donc l'état du système après la mesure est la projection normée de $|\psi\rangle$ sur le sous-espace propre associé à a_n .

$$\frac{\hat{P}_n |\psi\rangle}{\sqrt{\langle \psi | \hat{P}_n | \psi \rangle}}$$

Où \hat{P}_n est l'opérateur projecteur.

Postulat 6 : (l'évolution des systèmes) L'évolution du vecteur d'état $|\psi(t)\rangle$ avec le temps est donnée par l'équation de Schrödinger :

$$i\hbar \frac{d}{dt} |\psi(t)\rangle = \hat{H}(t) |\psi(t)\rangle$$

Où $H(t)$ est l'observable associé à l'énergie totale du système [SSS94].

1.3 Notions de l'informatique quantique

1.3.1 Bit quantique

En informatique classique, un bit peut prendre une des deux valeurs 0 ou 1, ce qui représente la présence ou non du courant électrique. En informatique quantique, on trouve le bit quantique ou qubit qui correspond aux systèmes quantiques à deux états 0 et 1 (photon, atome, etc.).

Le terme qubit a été introduit par Schumacher [Sch95], un qubit peut prendre une des valeurs 0 ou 1. Cependant, le bit quantique peut prendre les deux valeurs en même temps ou une superposition des deux états. Les deux états sont représentés par $|0\rangle$ et $|1\rangle$. L'état d'un système quantique est représenté par un vecteur dans l'espace de Hilbert de dimension 2, cet état peut être exprimé comme suit :

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \quad (1.2)$$

Où α et β sont des nombres complexes tels que $|\alpha|^2 + |\beta|^2 = 1$.

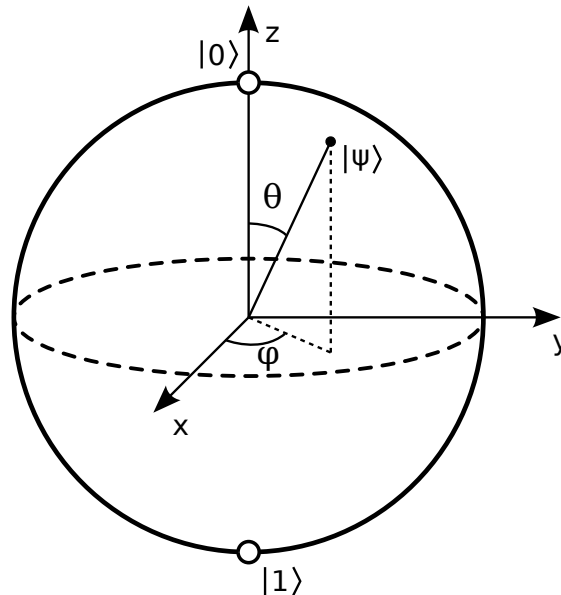


FIGURE 1.1: La sphère de Bloch

Le qubit est dans l'état 0 avec une probabilité $|\alpha|^2$ et dans l'état 1 avec une probabilité $|\beta|^2$. Il faut bien noter que l'opération de la mesure va détruire l'état initial de qubit, après la mesure l'état va correspondre au résultat de la mesure. Un qubit peut être représenté

par une forme matricielle :

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} , \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (1.3)$$

ou par la matrice de densité :

$$|0\rangle = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} , \quad |1\rangle = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \quad (1.4)$$

L'état d'un qubit peut être représenté aussi géométriquement par la sphère de Bloch (Figure 1.1).

Bit Classique	Bit Quantique
Un bit classique a toujours une valeur définie	On ne peut pas savoir la valeur de l'état d'un qubit sans mesure
Un bit peut avoir la valeur 1 ou 0	Un qubit peut être dans une superposition de 1 et 0 en même temps
Le bit peut être copié sans détruire son état initial	Impossible de copier sans erreur un qubit dans un état inconnu
Un bit peut être lu sans affecter sa valeur	Lire un qubit dans une superposition changera sa valeur
Lire un bit n'affecte pas un autre	Lire un qubit affectera l'autre, si le qubit lu est enchevêtré avec un autre qubit

Tableau 1.1: Comparaison entre le bit classique et le bit quantique

1.3.2 Réalisation physique d'un bit quantique

1.3.2.1 États internes d'un atome

La réalisation d'un qubit en utilisant un atome est faite en associant chaque niveau d'énergie de l'atome par un état de qubit [Ler13]. Dans la figure 1.2 on remarque les deux niveaux de l'atome :

- **Le niveau fondamental** : le niveau de basse énergie E_g , l'état quantique pour ce niveau est noté par $|g\rangle$ (ground state).
- **Le premier niveau excité** : le niveau d'énergie de l'atome en état excité E_e , l'état quantique pour ce niveau est noté par $|e\rangle$.

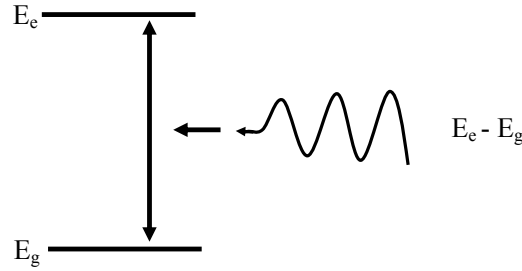


FIGURE 1.2: Atome à deux niveaux

1. Pour que l'atome passe de l'état fondamental à l'état excité $|g\rangle \rightarrow |e\rangle$, nous envoyons vers l'atome un photon d'une énergie égale à $E_e - E_g$, le photon est ensuite absorbé par l'atome.
2. Après une durée dans l'état excité l'atome revient à l'état fondamental $|e\rangle \rightarrow |g\rangle$ en émettant un photon d'une énergie de $E_e - E_g$, la durée de vie de l'état excité est généralement de quelques nanosecondes.
3. Si nous envoyons un photon à un atome dans le niveau excité, l'atome revient à l'état fondamental $|e\rangle \rightarrow |g\rangle$ en émettant un photon d'une énergie de $E_e - E_g$.

Si on répète d'une manière continue 1 et 3 avec une radiation résonante, l'état de l'atome va balancer entre $|g\rangle$ et $|e\rangle$, à un instant t l'état sera dans une superposition des deux états $|g\rangle$ et $|e\rangle$.

$$|\psi\rangle = \cos(\omega t/2) |g\rangle + \sin(\omega t/2) e^{i\phi} |e\rangle \quad (1.5)$$

Nous remarquons que par l'association des états $|g\rangle$ et $|e\rangle$ avec les états $|0\rangle$ et $|1\rangle$ respectivement, on peut placer l'atome dans n'importe quel état en changeant le temps de radiation. L'opération de la mesure est faite par la méthode suivante, on envoie une impulsion laser sur l'atome, si le photon est absorbé alors l'état de l'atome est $|g\rangle$ sinon l'atome est dans l'état $|e\rangle$ [Ler13].

1.3.2.2 Polarisation d'un photon

La lumière est l'ensemble des ondes électromagnétiques visibles par l'œil humain, dont les longueurs d'ondes sont entre 380 nm (violet) et 780 nm (rouge) [CoI87]. La lumière peut être représentée mathématiquement par un champ vectoriel transverse [Ler13], sur le référentiel (o, e_x, e_y, e_z) de coordonnées (x, y, z) , tel que le sens de propagation des ondes est sur z .

La polarisation représente le caractère vectoriel des vibrations lumineuses. Elle peut être remarquée à l'aide des cristaux ayant la propriété de biréfringence, et l'envoi d'un faisceau d'intensité I sur une lame biréfringente polarisée linéairement suivant une direction qui fait un angle θ avec l'axe du cristal. Le résultat est la division du faisceau en un faisceau

polarisé suivant Ox d'intensité $I \cos^2 \theta$ et un autre faisceau polarisé suivant Oy d'intensité $I \sin^2 \theta$.

En mécanique quantique, la lumière est un flot discret de photons. Un photon représente un objet quantique, donc on peut associer un état quantique à un vecteur de base de polarisation, $|x\rangle$ pour l'état de polarisation selon l'axe Ox , et $|y\rangle$ pour l'état de polarisation selon l'axe Oy .

$$|\theta\rangle = \cos(\theta) |x\rangle + \sin(\theta) |y\rangle \quad (1.6)$$

Selon le postulat de la mesure, si N est le nombre total des photons qui traverse la lame biréfringente, le nombre de photons sortants polarisés suivant Ox est $N_x = N \cos^2 \theta$ et le nombre de photons sortants polarisés selon Oy est $N_y = N \sin^2 \theta$ (Figure 1.3).

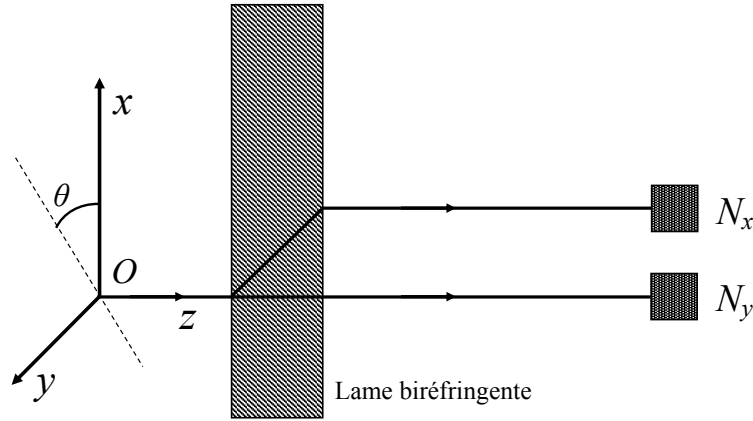


FIGURE 1.3: Décomposition de la polarisation par une lame biréfringente

Nous pouvons associer les états de qubit $|0\rangle$ et $|1\rangle$ à chacun des deux états de polarisation du photon $|x\rangle$ et $|y\rangle$ respectivement. On peut avoir n'importe quel état de superposition quantique par le changement de type de polarisation et l'emplacement du polariseur.

1.3.3 Registre quantique

Un registre quantique est un système quantique composé d'un ensemble de qubits, une superposition arbitraire de n qubits [Bla02]. Nous pouvons décrire l'état d'un registre quantique de n qubits par :

$$|\psi\rangle = \sum_{x=0}^{2^n-1} C_x |X\rangle \quad (1.7)$$

Où C_x représente les amplitudes de l'état de base $|X\rangle$ satisfont la propriété :

$$\sum_{x=0}^{2^n-1} |C_x|^2 = 1 \quad (1.8)$$

Un registre classique de n bits représente une seule valeur parmi les 2^n valeurs possibles, en revanche grâce au phénomène de la superposition d'états un registre quantique peut contenir les 2^n valeurs possibles en même temps.

On peut constater à partir de tableau 1.2 la puissance de l'informatique quantique qui dépasse exponentiellement l'informatique classique.

Nombre de qubits	Nombre de bits classique nécessaires pour obtenir la même puissance
10	1024
20	1 048 576
30	1 073 741 824
300	Plus que le nombre d'atomes dans l'univers visible

Tableau 1.2: Comparaison entre le nombre de qubits et le nombre équivalent des bits classiques

1.3.4 Principes de l'informatique quantique

L'informatique quantique est fondée sur cinq principes de base hérités de la mécanique quantique : la superposition d'états, l'interférence, l'enchevêtrement, le non-déterminisme et la non-duplication.

- **Superposition** : Un bit quantique peut être dans l'état 0 ou 1, mais il peut être aussi dans les deux états en même temps, c'est la propriété de superposition qui caractérise les systèmes quantiques. De la même manière un registre quantique de n qubits peut être dans une superposition de 2^n états à la fois [Fit07], ce qui donne une grande capacité de calcul et stockage aux ordinateurs quantiques.
- **Interférence** : Le phénomène d'interférence en informatique quantique, est l'opération d'augmenter (interférence constructive) ou diminuer (interférence destructive) l'amplitude d'un état, et par conséquent la probabilité de mesure [NC10].
- **Enchevêtrement** : Une des propriétés les plus importantes dans l'informatique quantique. Mathématiquement, l'état global d'un système quantique est dit enchevêtré, s'il est impossible de le diviser en un produit tensoriel de systèmes plus simples.

$$|\Phi_{AB}\rangle \neq |\Phi_A\rangle \otimes |\Phi_B\rangle \quad (1.9)$$

Autrement dit, deux qubits sont enchevêtrés si on trouve une corrélation entre les deux, et la modification de l'un affecte l'autre. Par exemple $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

représente une situation d'enchevêtrement car si le premier qubit est mesuré en i alors le deuxième qubit est sans aucun doute i , où $i = (0, 1)$.

Parfois, la modification d'une particule peut affecter en même temps l'état d'une autre particule à un autre endroit différent, c'est le principe de non-localité ou la téléportation dans la théorie quantique.

- **Non-déterminisme** : La mécanique quantique est indéterministe avec un comportement parfois aléatoire et imprévisible, où les mêmes causes ne produisent pas forcément les mêmes résultats. En informatique quantique, la valeur d'un qubit dans une superposition n'est pas connue avant l'observation, on ne peut jamais prédire le résultat de la mesure, mais on peut augmenter la probabilité d'avoir un état en utilisant les portes quantiques.
- **Non-clonage** : Le clonage d'un état n'est pas toujours possible en théorie quantique pour la raison que l'opération de mesure quantique détruit l'état initial de qubit, en conséquence on ne peut pas connaître avec certitude cet état, ce qui rend impossible de copier cet état indéterminé.

1.3.5 Mesure quantique

En informatique quantique, un qubit peut prendre l'état 0 ou 1, ou une superposition des deux états à la fois [Pre98]. La mesure quantique est une évaluation probabiliste d'un qubit ou d'un registre quantique afin d'avoir comme résultat une des valeurs superposées. L'observation de l'état d'un qubit est le résultat de l'évaluation de $|\alpha|^2$ et $|\beta|^2$, le qubit est dans l'état 0 avec une probabilité de $|\alpha|^2$, est dans l'état 1 avec une probabilité $|\beta|^2$ (voir figure 1.4).

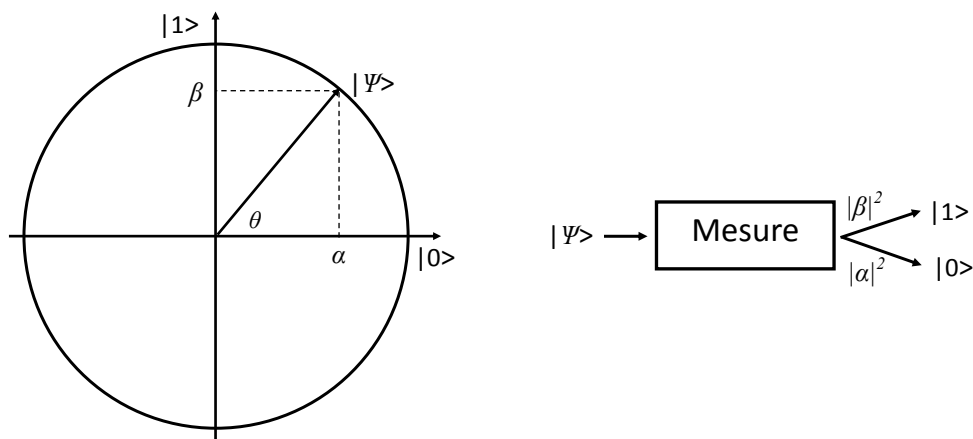


FIGURE 1.4: L'observation (mesure) quantique

Mathématiquement, la mesure est une projection de l'état d'un système quantique sur une des bases de l'espace de Hilbert. Selon le principe de non-déterminisme de l'informatique

quantique la mesure de la même superposition d'état ne donne pas toujours le même résultat [RP00].

1.3.6 Calcul quantique

Le calcul dans un ordinateur quantique est réalisé à l'aide d'un ensemble des opérations logiques dans un espace de Hilbert de 2 dimensions, le calcul quantique n'est qu'une évolution d'un système quantique à partir d'un vecteur d'état initial $|\psi(0)\rangle$ par un opérateur unitaire d'évolution $U(t)$, et à la fin la mesure de l'état de système quantique pour obtenir le résultat du calcul.

$$|\psi(t)\rangle = U(t) |\psi(0)\rangle \quad (1.10)$$

Tout le processus d'évolution est purement quantique dans l'espace de Hilbert, cette évolution unitaire est réversible donc $U^{-1}(t, t_0) = U(t_0, t)$ [LB03].

La manipulation de système quantique est réalisée par des opérateurs d'évolutions unitaires (portes quantiques) ou par une combinaison des portes quantiques (circuits quantiques).

1.3.7 Portes quantiques

Une porte quantique est une opération unitaire agissant sur un ou plusieurs qubits afin de permettre le traitement de l'information quantique. Il existe un nombre infini de portes quantiques [Lac07], nous présentons ici quelques exemples :

1.3.7.1 Portes quantiques de Pauli

Un ensemble des opérations quantiques simples applicables sur un seul qubit peut être représenté par des matrices unitaires de 2x2 [Lév04] :

- **L'identité (I) :**

$$|0\rangle \rightarrow |0\rangle$$

$$|1\rangle \rightarrow |1\rangle$$

La matrice correspondante :

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

- **La négation (X) :**

$$|0\rangle \rightarrow |1\rangle$$

$$|1\rangle \rightarrow |0\rangle$$

La matrice correspondante :

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

- **La transformation de décalage de phase (Z) :**

$$|0\rangle \rightarrow |0\rangle$$

$$|1\rangle \rightarrow |1\rangle$$

La matrice correspondante :

$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

- **La combinaison (Y) de X et Z, tel que $(Y) = ZX$:**

$$|0\rangle \rightarrow |1\rangle$$

$$|1\rangle \rightarrow |0\rangle$$

La matrice correspondante :

$$\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

1.3.7.2 Portes U-Contrôlé

En informatique quantique, on peut utiliser des actions qui garantissent le contrôle de l'état d'un ou plusieurs qubits, et ensuite suivant l'état des qubits de contrôles, nous pouvons effectuer ou non le changement sur les autres qubits. On appelle $\Lambda^k U$ U-contrôlé à k qubits de contrôles et n qubits de données cibles (Figure 1.5).

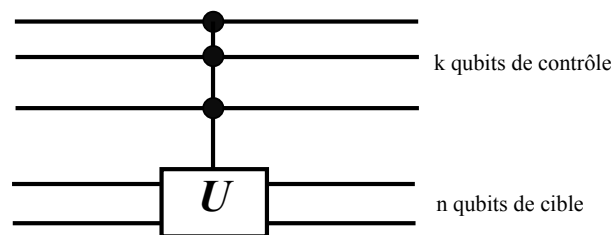


FIGURE 1.5: La porte quantique U-Contrôlé $\Lambda^k U$

- **Porte Controlled-Not CNOT ($\Lambda^1 X$) :** Une porte U-contrôlé avec un qubit de contrôle, si le premier qubit est 1, le deuxième qubit est inversé, si le premier qubit est 0, on ne fait rien. Dans la figure 1.6 le cercle noir représente le qubit de contrôle, et le cercle ouvert représente la négation conditionnelle, la porte Controlled-Not

peut avoir plusieurs qubits de contrôles.

$$\begin{aligned} |00\rangle &\rightarrow |00\rangle \\ |01\rangle &\rightarrow |01\rangle \\ |10\rangle &\rightarrow |11\rangle \\ |11\rangle &\rightarrow |10\rangle \end{aligned}$$

La matrice correspondante :

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

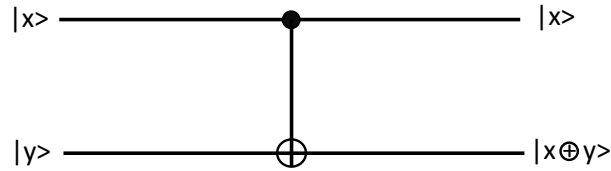


FIGURE 1.6: La porte quantique Controlled-Not

- **Porte de Toffoli Controlled-Controlled-Not ($\Lambda^2 X$)** : Une porte logique avec deux qubits de contrôles et une négation conditionnelle, si les deux qubits de contrôles sont à 1 le troisième qubit est inversé, sinon on ne fait rien. Cette porte est réversible et universelle (Figure 1.7).

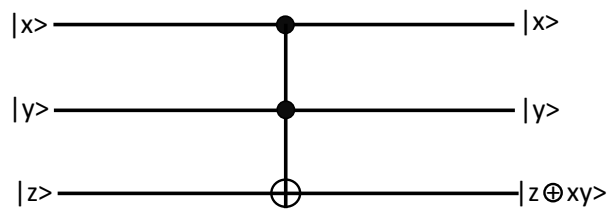


FIGURE 1.7: La porte quantique Toffoli

1.3.7.3 Porte racine-carré-du-not

L'application de la porte $U_{\sqrt{NOT}}$ deux fois successivement donne l'opérateur U_{NOT} , dans l'informatique classique on ne trouve pas une porte pareille, en raison de l'absence d'un état intermédiaire entre les valeurs 0 et 1 (Figure 1.8).

$$U_{\sqrt{NOT}}U_{\sqrt{NOT}}|0\rangle = U_{NOT}|0\rangle = |1\rangle$$

$$U_{\sqrt{NOT}}U_{\sqrt{NOT}}|1\rangle = U_{NOT}|1\rangle = |0\rangle$$

$$U_{\sqrt{NOT}}|0\rangle = \left(\frac{1}{2} + \frac{i}{2}\right)|0\rangle + \left(\frac{1}{2} - \frac{i}{2}\right)|1\rangle$$

$$U_{\sqrt{NOT}}|1\rangle = \left(\frac{1}{2} - \frac{i}{2}\right)|0\rangle + \left(\frac{1}{2} + \frac{i}{2}\right)|1\rangle$$

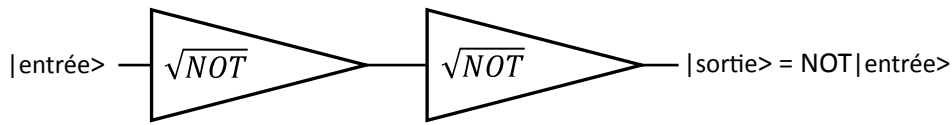


FIGURE 1.8: La porte quantique racine-carré-du-not

1.3.7.4 Porte Walsh-Hadamard

Une transformation unitaire qui permet d'effectuer une rotation à l'état d'un qubit :

$$U_{WH}|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

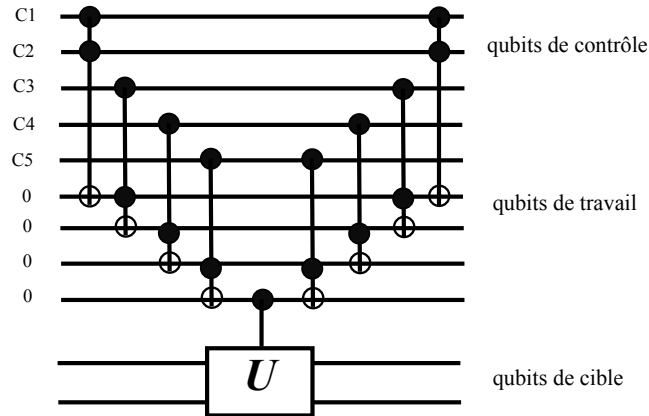
$$U_{WH}|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

Cette porte est utilisée en général pour créer des superpositions d'états, elle permet de créer 2^n valeurs à partir de n qubits.

1.3.8 Circuits quantiques

L'informatique quantique doit fournir les éléments de base nécessaires pour la réalisation d'un ordinateur quantique, les circuits quantiques sont parmi les éléments les plus importants. Un circuit quantique est une combinaison de plusieurs portes quantiques dont le but est de remplir une fonction plus compliquée [Dum02].

Nous prenons ici un exemple (Figure 1.9) du circuit $\Lambda^5 U$ qui est réalisé à l'aide de 8 portes Toffoli et une porte CNOT.


 FIGURE 1.9: Le circuit quantique $\Lambda^5 U$

1.3.9 Cryptographie quantique

La transmission des messages secrets est parmi les plus anciens problèmes. La solution est de trouver une méthode pour coder le message d'une manière que seulement un destinataire avec la clé puisse le déchiffrer, ce qui rend impossible pour un espion qui ne possède pas la clé. Traditionnellement, on appelle Alice et Bob les deux parties de la communication et Eve comme l'espionne. Il y a typiquement deux méthodes de codage[Ler13] :

- **Le codage à clé privée** : La clé de codage et décodage est partagée entre Alice et Bob pour qu'ils puissent échanger leurs messages en toute sécurité, alors que c'est impossible pour Eve qui ne possède pas la clé. Le problème avec cette méthode réside dans l'échange de la clé de codage sans qu'elle soit interceptée.
- **Le codage à clé publique** : Alice transmet la clé de codage publiquement à Bob, mais ne lui transmet pas la clé de décodage, donc Bob n'est capable que de coder les messages, puis il transmet les messages à Alice pour les déchiffrer. Le point faible de cette méthode se situe dans la capacité d'un espion de trouver la clé de décodage à partir de la clé de codage.

En cryptographie quantique, on utilise le premier protocole (le codage à clé privée), elle est fondée sur l'idée que l'opération de mesure quantique va modifier l'état de l'objet quantique et en conséquence le message, de ce fait il est impossible pour un espion d'intercepter un message sans être détecté, de cette façon nous sommes sûrs que la clé est sécurisée.

L'information va être transportée à l'aide des photons et le codage des bits est réalisé par la polarisation selon les deux bases \oplus (polarisation horizontale-verticale) et \otimes (polarisation

Alice	Bits à Envoyer	0	0	1	1	0	0
Alice	Choix de base	\oplus	\oplus	\otimes	\oplus	\oplus	\otimes
Alice	Polarisation Envoyée	$ \rightarrow\rangle$	$ \rightarrow\rangle$	$ \nearrow\rangle$	$ \uparrow\rangle$	$ \rightarrow\rangle$	$ \searrow\rangle$
Bob	Choix de base	\oplus	\otimes	\otimes	\oplus	\otimes	\oplus
Bob	Polarisation mesurée	$ \rightarrow\rangle$	$ \nearrow\rangle$	$ \nearrow\rangle$	$ \uparrow\rangle$	$ \searrow\rangle$	$ \rightarrow\rangle$
Bob	Bits lus	0	1	1	1	0	0
Alice et Bob	Bits acceptés	oui	non	oui	oui	non	non
Alice et Bob	Clé	0		1	1		

Tableau 1.3: La composition d'un message secret

inclinée à 45°) de cette manière :

$$\begin{aligned} |\uparrow\rangle &\rightarrow 1 \\ |\rightarrow\rangle &\rightarrow 0 \\ |\nearrow\rangle &\rightarrow 1 \\ |\searrow\rangle &\rightarrow 0 \end{aligned}$$

Notons bien que un photon polarisé selon la base \oplus est ensuite analysé selon la base \otimes va être mesuré par une probabilité de $1/2$, alors que s'il est analysé selon la même base il va être mesuré avec certitude.

$$\begin{aligned} |\nearrow\rangle &= \frac{1}{\sqrt{2}}(|\uparrow\rangle + |\rightarrow\rangle) \\ |\searrow\rangle &= \frac{1}{\sqrt{2}}(|\uparrow\rangle - |\rightarrow\rangle) \end{aligned}$$

Alice va coder le message en binaire en choisissant pour chaque bit une des deux bases de polarisation des photons \oplus et \otimes . Lors de la réception des photons, Bob analyse les photons aussi sur une des deux bases d'une manière aléatoire, et envoie publiquement à Alice la liste des analyseurs utilisés. Alice compare cette liste avec la sienne et communique la liste publiquement à Bob, l'intersection des deux listes ou bien les bits avec les mêmes valeurs pour Alice et Bob représente le message secret (Tableau 1.3).

Pour vérifier que le message n'a pas été écouté, il suffit de prendre un échantillon aléatoire de clé et communiquer publiquement, il doit être identique, une différence signifie que le message a été intercepté.

1.3.10 Correction d'erreurs quantiques

La décohérence est le problème principal du calcul quantique [Zur03], ce problème réside dans l'interaction des qubits avec l'environnement extérieur, donc une perte de l'information contenue dans le système quantique vers l'environnement. Nous citons ici quelques modèles d'erreurs :

- **Inversion de qubit (X) :**

$$|0\rangle \rightarrow |1\rangle$$

$$|1\rangle \rightarrow |0\rangle$$

- **Erreur de phase (Z) :**

$$|0\rangle \rightarrow |0\rangle$$

$$|1\rangle \rightarrow -|1\rangle$$

- **Inversion de qubit + erreur de phase (Y) :**

$$|0\rangle \rightarrow -i|1\rangle$$

$$|1\rangle \rightarrow i|0\rangle$$

Exemple d'une erreur (XIZ) :

$$\begin{aligned} |011\rangle &= |0\rangle \oplus |1\rangle \oplus |1\rangle \\ X \oplus I \oplus Z |011\rangle &= (X \oplus I \oplus Z)(|0\rangle \oplus |1\rangle \oplus |1\rangle) \\ &= (X|0\rangle) \oplus (I|1\rangle) \oplus (Z|1\rangle) \\ &= -|1\rangle \oplus |1\rangle \oplus |1\rangle \\ &= -|111\rangle \end{aligned}$$

Les codes quantiques ou les codes correcteurs sont des méthodes utilisées pour protéger l'information quantique contre les effets de la décohérence. L'état de superposition ($|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$) du système quantique doit être protégé, notons bien que la copie est impossible dans l'informatique quantique, et que la mesure modifie l'état de qubit.

Exemple d'un code correcteur d'erreurs à trois qubits :

La première étape est le codage de l'information sur 3 qubits, en rajoutant deux qubits redondants $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, après le codage on obtient $|\psi\rangle_C = \alpha|000\rangle + \beta|111\rangle$

Si il y a au plus une erreur de type (X) ou (bit-flip), il y aura trois possibilités :

$$X_1 \rightarrow |100\rangle + |011\rangle$$

$$X_2 \rightarrow |010\rangle + |101\rangle$$

$$X_3 \rightarrow |001\rangle + |110\rangle$$

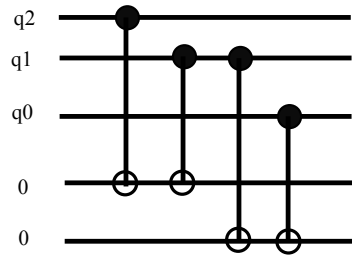


FIGURE 1.10: Un code correcteur à trois bits

Nous essayons de détecter l'erreur sans affecter l'état, nous ajoutons deux qubits auxiliaires dans l'état 0 comme dans le schéma (Figure 1.10), s'il y a une erreur, la mesure des qubits auxiliaires indique le qubit erroné. La correction est réalisée en appliquant un opérateur (X) sur le qubit en question.

1.3.11 Téléportation quantique

La téléportation quantique est une des applications les plus surprenantes de l'enchevêtrement quantique. On veut envoyer le contenu de l'état d'un qubit $|\psi_A\rangle$ du site A au site B (l'état de qubit $|\psi_A\rangle$ est inconnu pour A), A et B sont lointains mais ils partagent deux qubits $|\psi_B\rangle$ et $|\psi_C\rangle$ enchevêtrés. Le processus de la téléportation est comme suit :

Etape 1 : le site A applique un CNOT sur $(|\psi_A\rangle, |\psi_B\rangle)$.

Etape 2 : l'envoi de qubit sur la porte de Hadamard, l'état de qubit $|\psi_C\rangle$ est déterminé par celui du couple $(|\psi_A\rangle, |\psi_B\rangle)$ grâce au phénomène de l'enchevêtrement de $(|\psi_B\rangle, |\psi_C\rangle)$.

Etape 3 : le site A mesure l'état de $(|\psi_A\rangle, |\psi_B\rangle)$ pour transmettre les qubits.

1.3.12 Critères de la réalisation d'un ordinateur Quantique

Les critères nécessaires pour l'implémentation d'un ordinateur quantique sont définis par DiVincenzo [D⁺00], ces critères définissent le standard dans les recherches sur l'implémentation des ordinateurs quantiques.

- **Qubits** : l'ordinateur quantique doit avoir des qubits.
- **Initialisation** : l'initialisation du registre quantique dans un état connu doit être faisable, par exemple $|0\rangle = |000\dots000\rangle$.
- **Calcul** : la possibilité de manipulation d'un registre quantique par un ensemble universel de portes logiques.
- **Cohérence** : l'interaction des qubits avec l'environnement externe doit être minimisée pendant le calcul, afin d'éviter les erreurs liées à la décohérence.
- **Mesure** : chaque qubit doit pouvoir subir une mesure projective, et la valeur du bit classique trouvée doit être obtenue avec une grande efficacité.

1.3.13 Algorithmes quantiques

Un algorithme quantique est une suite des opérations quantiques unitaires réalisées par des portes logiques quantiques. La logique quantique permet de réaliser les calculs de façon plus rapide par rapport à l'informatique classique [Fey82]. Cette puissance vient du principe de la superposition quantique où on a la possibilité de représenter 2^n valeurs dans un seul registre quantique de n qubits, et en conséquence un traitement massivement parallèle.

L'exploitation du principe de la superposition et d'intrication de qubits nous permet de rendre certains problèmes classiques difficiles à des problèmes plus simples en utilisant l'informatique quantique. Les algorithmes quantiques sont de ce fait plus efficaces que les algorithmes classiques, bénéficiant des capacités de l'informatique quantique.

Plusieurs algorithmes quantiques ont été développés afin de profiter de la puissance de l'informatique quantique à la résolution des problèmes complexes connus, parmi ses algorithmes nous avons l'algorithme de Shor [Sho94] pour la factorisation de grands nombres, et celui de Grover [Gro96] pour la recherche dans une base de données non triée.

1.3.13.1 Algorithme de Shor

L'algorithme de Shor [Sho94] est un algorithme quantique pour factoriser un nombre entier de n bits en $O(N^2 \ln N \ln(\ln(N)))$ opérations, alors que le meilleur algorithme de

factorisation classique demande $O(\exp(\frac{64}{9})^{\frac{1}{3}} N^{\frac{1}{3}} (\ln(N))^{\frac{2}{3}})$, cet algorithme repose sur la transformation de Fourier quantique afin d'accélérer le calcul.

L'algorithme est divisé en deux parties, la première partie consiste à transformer le problème de factorisation en un problème de recherche de période d'une fonction, ici les méthodes classiques peuvent être utilisées, la deuxième partie consiste à trouver la période en utilisant la transformée de Fourier quantique pour accélérer le traitement.

Les étapes de l'algorithme de Shor : les étapes 1, 2, 3 et 10 peuvent être exécutées sur un ordinateur classique.

Etape 1 : Si le nombre N est premier, pair ou un exposant d'un nombre premier, dans ce cas nous utilisons les algorithmes classiques qui sont efficaces pour résoudre le problème.

Etape 2 : Choisir un entier q qui est une puissance de 2 tel que $N^2 \leq q \leq 2N^2$.

Etape 3 : Soit X un nombre aléatoire, qui est co-premier avec N , on dit co-premier si leur plus grand diviseur commun est 1.

Etape 4 : Diviser le registre quantique en deux, Reg1 et Reg2, afin de représenter les nombres X et N sur Reg1 et Reg2 respectivement.

Etape 5 : Préparer le Reg1 dans une superposition des entiers de 0 à $q-1$ de même amplitudes, et charger Reg2 par des 0 pour avoir l'état quantique :

$$\frac{1}{\sqrt{q}} \sum_{a=0}^{q-1} |a, 0\rangle \quad (1.11)$$

Etape 6 : On utilise le principe de la superposition pour calculer $X^{|a\rangle} \bmod N$, et placer les résultats dans Reg2, l'état quantique sera :

$$\frac{1}{\sqrt{q}} \sum_{a=0}^{q-1} |a, X^a \bmod N\rangle \quad (1.12)$$

Etape 7 : La mesure de Reg2 et K :

$$\frac{1}{\sqrt{|A|}} \sum_{a \in A} |a, K\rangle \quad (1.13)$$

où A est l'ensemble des a tel que $X^a \bmod N = K$ et $|A|$ le nombre d'éléments dans cet ensemble.

Etape 8 : Calculer la transformée de Fourier discrète sur Reg1 :

$$\frac{1}{\sqrt{\|A\|}} \sum_{a \in A} \frac{1}{\sqrt{q}} \sum_{c=0}^{q-1} |c, K\rangle . e^{2\pi ac/q} \quad (1.14)$$

Etape 9 : La mesure de l'état de Reg1 donnera l'entier m , qui est avec une grande probabilité un multiple de $\frac{q}{r}$, où r est la période recherchée.

Etape 10 : Calculer r en se basant sur m et q .

Etape 11 : Le facteur de N est calculé par le $PGCD(X^{r/2} + 1, N)$ et $PGCD(X^{r/2} - 1, N)$. Si le facteur de N est trouvé alors on arrête, sinon on reprend à partir de l'étape 4.

L'algorithme de Shor peut échouer dans deux cas, le résultat de la transformée de Fourier est 0, ou on obtient comme résultat de facteur la valeur 1 ou N .

1.3.13.2 Algorithme de Grover

L'algorithme de Grover [Gro96] permet de trouver les éléments X satisfaisant une condition P dans une liste non structurée de taille N , cet algorithme à une complexité de $O(\sqrt{N})$, alors que les méthodes classiques ont une complexité de $O(N)$. L'algorithme de Grover peut être utilisé pour la résolution de certains problèmes de type NP-complet.

Soit $P(x)$ la fonction utilisée par l'algorithme classique, U_p l'opération quantique qui implémente la fonction classique (la fonction oracle).

$$U_p : |x, 0\rangle = |x, P(x)\rangle \quad (1.15)$$

Etape 1 : La création d'une superposition sur un registre de taille N initialisé à $|00\dots 00\rangle$, cette superposition représente toutes les solutions possibles.

$$|\psi\rangle = \frac{1}{\sqrt{2^N}} \sum_{x=0}^{2^N-1} |x\rangle \quad (1.16)$$

Etape 2 : Pour chaque élément qui satisfait le critère P on applique un opérateur shift (inverser la phase).

Etape 3 : L'algorithme applique ensuite une opération d'augmentation d'amplitude, le résultat est l'amplification de l'état de la bonne solution, et diminue les autres états. L'algorithme de Grover utilise l'opérateur d'in-

versement par rapport au moyen D :

$$D = \begin{pmatrix} \frac{2}{N} - 1 & \frac{2}{N} & \cdots & \frac{2}{N} \\ \frac{2}{N} & \frac{2}{N} - 1 & \cdots & \frac{2}{N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{2}{N} & \frac{2}{N} & \cdots & \frac{2}{N} - 1 \end{pmatrix} \quad (1.17)$$

Etape 4 : Appliquer une opération de mesure sur le registre quantique pour extraire la bonne solution.

Chaque itération augmente l'amplitude de la bonne solution et les amplitudes des autres solutions seront diminuées, le nombre d'itérations optimal est de $\frac{\pi}{4}\sqrt{2^N}$, la mesure du registre quantique après ce nombre d'itérations donne la bonne solution mais avec un taux d'échec de 2^{-N} .

1.3.14 Ordinateurs quantiques

1.3.14.1 Historique

L'idée derrière l'ordinateur quantique revient à Richard Feynman [Fey67], dans les années 70 Feynman a proposé l'utilisation des concepts de l'informatique quantique, afin de produire un ordinateur plus puissant que l'ordinateur classique. Ce projet n'a pas retenu l'attention, et pendant des décennies les physiciens ont douté qu'un ordinateur quantique puisse exister.

En 1994 Peter Shor [Sho94], chercheur chez AT & T, développe un algorithme quantique de factorisation des grands nombres, l'algorithme de Shor a apporté une amélioration exponentielle par rapport aux méthodes classiques.

En 1996, Lov Grover [Gro96], de Bell Labs, développe un algorithme quantique qui permet de trouver un élément dans une liste non structurée.

En 1996, Neil Gershenfeld et Isaac Chuang, inventent une nouvelle approche réalisable de l'ordinateur quantique, où l'information est représentée par les spins nucléaires des atomes contrôlés par des techniques de résonance magnétique nucléaire.

En 1998, le premier ordinateur quantique de 2 qubits est présenté par IBM.

En 1999, la démonstration de l'algorithme quantique de Grover sur un ordinateur quantique à 3-qubits de IBM.

En 2001, IBM crée un calculateur quantique de 7 qubits de molécules de chloroforme.

La société D-Wave a annoncé officiellement le 13 février 2007 avoir réalisé un ordinateur quantique à base solide de 16 qubits.

En 2010, l'université de Bristol annonce la création d'un processeur quantique optique, en silicium, capable d'exécuter l'algorithme de Shor.

1.3.14.2 Réalisation de l'ordinateur quantique

Le défi dans le développement d'un ordinateur quantique réside dans la réalisation physique des qubits. Le qubit est l'élément essentiel dans le traitement de l'information par un ordinateur quantique. Le qubit est un élément quantique à deux niveaux, qui peut être dans une superposition des deux états en même temps selon les lois de la mécanique quantique. Notons bien que les opérations sur les qubits doivent être faites en utilisant les opérations quantiques aussi.

La décohérence représente aussi un des plus grands problèmes dans l'informatique quantique. La décohérence quantique peut être définie par l'incapacité d'un qubit de préserver son état quantique toute la période du calcul [Sch05], sans interagir avec l'environnement extérieur. Le résultat de la décohérence est une perte de l'information de qubits [Sho95]. Il existe actuellement des méthodes qui permettent la correction des erreurs engendrées par la décohérence, mais ces méthodes ne sont efficaces que si le taux d'erreurs est faible.

Plusieurs méthodes basées sur la physique quantique ont été proposées afin de réaliser un ordinateur quantique, nous citons ici quelques méthodes :

- **La résonance magnétique nucléaire** : C'est le système qui a prouvé les meilleurs résultats jusqu'à ce jour (ordinateur quantique de 7 qubits de IBM en 2001), les qubits sont les états de spin nucléaire d'atomes dans une molécule. La manipulation des spins est effectuée par la fréquence de résonance qui est différente pour chaque spin.
- **Polarisation des photons** : Les deux axes de polarisation des photons représentent les deux états de qubit, les opérations quantiques sont effectuées par les techniques d'interférométrie.
- **Les ions piégés** : Dans cette technique, les deux niveaux d'énergie d'une particule représentent les états d'un qubit, la modification de l'état des qubits est réalisée à l'aide des impulsions laser.
- **Ordinateur quantique de Kane** : Utilise des spins nucléaires d'atomes donneurs de phosphore dans un cristal de silicium isotopique pur, le spin du noyau atomique forme un système à deux niveaux donc les deux états de qubit, un électron piégé sur l'atome de phosphore est utilisé pour contrôler la fréquence de résonance du

noyau.

- **Les qubits solides :** Basé sur la physique du solide comme les ordinateurs actuels, une puce en silicium utilise trois nanojonctions Josephson, deux jonctions servent comme un qubit, la troisième est utilisée comme un instrument de mesure.

1.4 Conclusion

Dans ce chapitre, nous avons commencé par une présentation de la mécanique quantique et les principes de base de cette nouvelle théorie, ainsi que le formalisme mathématique et les lois qui contrôlent cette théorie. Ensuite, nous avons présenté les notions de base et les concepts de l'informatique quantique, ainsi que les applications possibles de cette technique. Enfin, nous avons exposé les différentes méthodes de la réalisation de ces concepts en utilisant les techniques de la physique quantique comme la polarisation de photons et la résonance magnétique nucléaire.

L'informatique quantique est un domaine multidisciplinaire qui fait appel à plusieurs domaines tels que la mécanique quantique, l'informatique et la mathématique, cette collaboration interdisciplinaire a pour objectif la réalisation d'un ordinateur quantique capable de profiter de toute la puissance théorique de l'informatique quantique.

Un ordinateur quantique fonctionnel est encore loin d'être réalisé, cependant l'utilisation de quelques concepts inspirés de la théorie quantique sur un ordinateur classique est possible. Dans le chapitre suivant nous allons présenter une des approches hybrides, nommée les algorithmes évolutionnaires quantiques.

Algorithmes évolutionnaires quantiques

Nous présentons dans ce chapitre les algorithmes évolutionnaires inspirés du quantique qui sont fondés sur la combinaison des algorithmes évolutionnaires et l'informatique quantique. Nous exposons ensuite les différents types des algorithmes évolutionnaires quantiques et leurs avantages par rapport aux algorithmes évolutionnaires classiques.

2.1 Introduction	29
2.2 Algorithmes évolutionnaires	29
2.2.1 Classes des algorithmes évolutionnaires	30
2.2.2 Principes des algorithmes évolutionnaires	31
2.3 Principes des algorithmes évolutionnaires quantiques	34
2.3.1 Représentation quantique dans les AEQs	35
2.3.2 Observation dans les AEQs	36
2.3.3 Opérations de mise à jour	37
2.3.4 Structure générale des AEQs	39
2.3.5 Types des AEQs	41
2.3.6 Caractéristiques des AEQs	42
2.3.7 Renforcement et Diversification dans les AEQs	43
2.3.8 Avantages des AEQs	44
2.4 Conclusion	44

2.1 Introduction

Le domaine de l'informatique quantique est encore dans ses débuts, mais il présente déjà beaucoup des possibilités et des idées innovantes grâce à sa théorie solide, et ses concepts extraordinaires, comme la superposition et l'observation probabiliste, mais l'exploitation des vraies capacités de ce domaine ne sera possible qu'après la réalisation d'un ordinateur quantique.

La mise en place d'un ordinateur quantique capable de réaliser les différentes opérations quantiques, et de préserver sa cohérence pour une longue durée n'est pas encore possible, et par conséquent l'implémentation des algorithmes quantiques purs représente une tâche très difficile. Ce qui a mené les chercheurs de penser à des alternatives pour faire face à ce problème.

Plusieurs algorithmes sont apparus dernièrement, qui peuvent être appliqué sur des ordinateurs classiques, mais en même temps ils sont fondés sur les concepts de l'informatique quantique, parmi ces algorithmes on trouve les algorithmes évolutionnaires inspirés du quantique, que nous allons présenter dans ce chapitre afin d'illustrer leurs puissances et leurs supériorités par rapport aux algorithmes évolutionnaires classiques.

2.2 Algorithmes évolutionnaires

En informatique, on est confronté à des problèmes que l'espace de recherche a une taille et une complexité importante, que les méthodes classiques de recherche d'une solution optimale en un temps raisonnable est impossible. De là, l'idée de trouver des techniques permettant la localisation rapide des solutions sous-optimales en un temps raisonnable est utile [Spa04].

Les algorithmes évolutionnaires sont des méthodes d'optimisation stochastique basées sur le concept de l'évolution naturelle des populations (la théorie Darwinienne)[Ber10]. Ils peuvent être définis comme des techniques de programmation fondées sur le principe de l'évolution d'espèces, qui permettent aux individus les mieux adaptées à leur environnement de se reproduire [Mit98], en utilisant les mécanismes de la sélection naturelle génétique : croisement, mutation, sélection, etc.

Le succès des algorithmes évolutionnaires revient essentiellement à la simplicité de la mise en œuvre et leurs puissances dans la résolution des problèmes complexes [YSB08]. Ces méthodes de programmation donnent l'impression qu'on a rajouté la capacité d'innovation et la créativité à l'informatique.

La difficulté dans le développement des algorithmes évolutionnaires réside dans le choix des

paramètres de l'algorithme, ce qui représente une tâche très délicate, de ce fait la réussite de la mise en œuvre d'une solution nécessite une bonne compréhension du problème à traiter.

2.2.1 Classes des algorithmes évolutionnaires

L'ensemble des algorithmes évolutionnaires est divisé en quatre grandes familles (voir figure 2.1). Ces classes ne sont différentes que sur quelques détails d'implantation des opérateurs et les procédures de la sélection et le remplacement des individus de la population [Ter06].

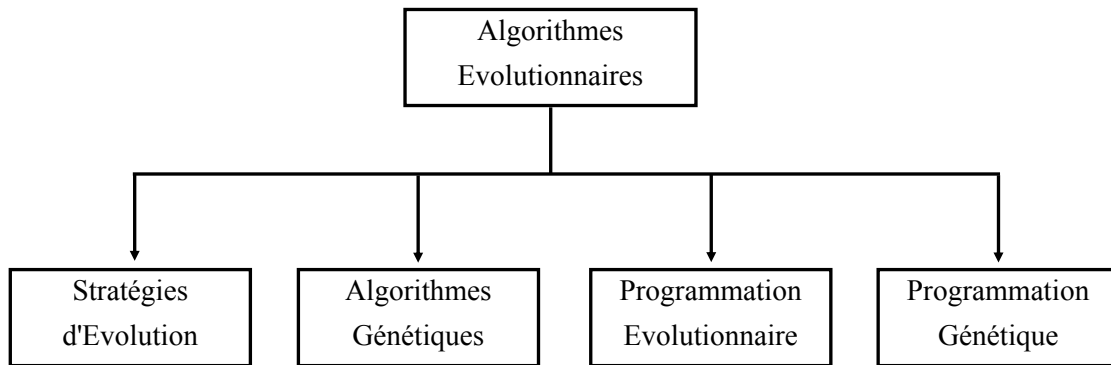


FIGURE 2.1: Les classes des algorithmes évolutionnaires

2.2.1.1 Stratégies d'évolution

Les stratégies d'évolution sont les premiers algorithmes évolutionnaires, inventées par Rechenberg dans les années 60 [Rec65], afin de résoudre les problèmes d'optimisations numériques dans un espace de paramètres réels. C'est un algorithme itératif qui manipule un ensemble des variables réelles, à l'aide des opérateurs de mutation et de sélection. Les stratégies d'évolution sont les meilleurs algorithmes pour les problèmes purement numériques.

- Les stratégies d'évolution utilisent une codification des paramètres du problème à résoudre en nombres réels.
- La mutation est généralement effectuée par l'ajout d'une valeur aléatoire, tirée au sein d'une distribution normale.
- La sélection est effectuée par une évaluation déterministe des meilleurs individus, selon la valeur de la fonction objectif.

2.2.1.2 Programmation évolutionnaire

Inventée par Fogel, Owens et Walsh en 1965 [BFM97], développée au début pour la découverte d'automates à états finis pour l'approximation des séries temporelles, en-

suite elle a été utilisée pour des espaces de recherches divers. Ce type d'algorithmes évolutionnaires est basé sur l'utilisation de l'opérateur de mutation, en appliquant un taux de mutation élevé aux mauvais individus et un taux faible aux bons individus [Ter06]. La représentation des individus en programmation évolutionnaire n'a pas la forme spécifique du génome.

2.2.1.3 Programmation génétique

Inventée par Koza en 1992 [Koz92], au début elle a été considérée comme un sous-domaine des algorithmes génétiques, après quelques avancées dans les années 2000, la programmation génétique est devenue une branche à part. Le but de la programmation génétique est de trouver des programmes qui représentent la meilleure solution à une tâche donnée.

Elle permet l'apprentissage automatique en utilisant les algorithmes évolutionnaires, afin d'optimiser une population de programmes. La programmation génétique cherche à atteindre un des rêves des développeurs, faire écrire le programme par un autre programme.

Les individus de la population sont des programmes représentant des solutions à un problème, qui seront exprimés sous la forme d'arbre, afin d'appliquer les opérateurs génétiques [Cra85].

2.2.1.4 Algorithmes génétiques

Développés dans les années 70 par Holland [Hol75] puis amélioré par Goldberg [GH88], les algorithmes génétiques sont des méthodes de recherche combinatoire, dont le but est de trouver une solution optimale à un problème d'optimisation [Mit98]. Les algorithmes génétiques sont la branche des algorithmes évolutionnaires la plus connue et la plus utilisée.

- Les algorithmes génétiques favorisent l'utilisation du croisement comme un opérateur de recherche.
- Les algorithmes génétiques utilisent la mutation, mais avec une faible probabilité.
- Les algorithmes génétiques utilisent une méthode de sélection de type probabiliste.
- Le codage des individus sous la forme de chaînes de bits, qui représente une description d'un point dans l'espace d'états.

2.2.2 Principes des algorithmes évolutionnaires

Les algorithmes évolutionnaires sont des techniques itératives qui utilisent des opérateurs de variation stochastiques sur un ensemble des individus formant une population. Les individus de la population représentent l'ensemble des solutions possibles (espace de re-

cherche) d'un problème donné.

- i. La première étape des algorithmes évolutionnaires est la création de la population initiale.
- ii. L'évaluation des individus, pour chaque individu dans la population, on associe une valeur de fitness qui représente la qualité de cette solution selon un ou plusieurs critères donnés.
- iii. La sélection, une opération de sélection permet aux individus avec les meilleures valeurs de fitness de se reproduire afin d'assurer la pérennité des meilleures caractéristiques, ces individus vont être recombinaés afin de créer à chaque génération de nouveaux individus mieux que les individus des générations précédentes.
- iv. À chaque itération de nouveaux individus sont générés à l'aide des opérations génétiques (croisement et mutation), en recombinaant les caractéristiques des individus parents, ces individus représentent les nouvelles solutions trouvées.
- v. L'évaluation de nouveaux individus.
- vi. Le remplacement des individus parents par les meilleurs individus descendants.
- vii. La fin du processus évolutionnaire après un certain nombre d'itérations, ou si la performance recherchée est obtenue.

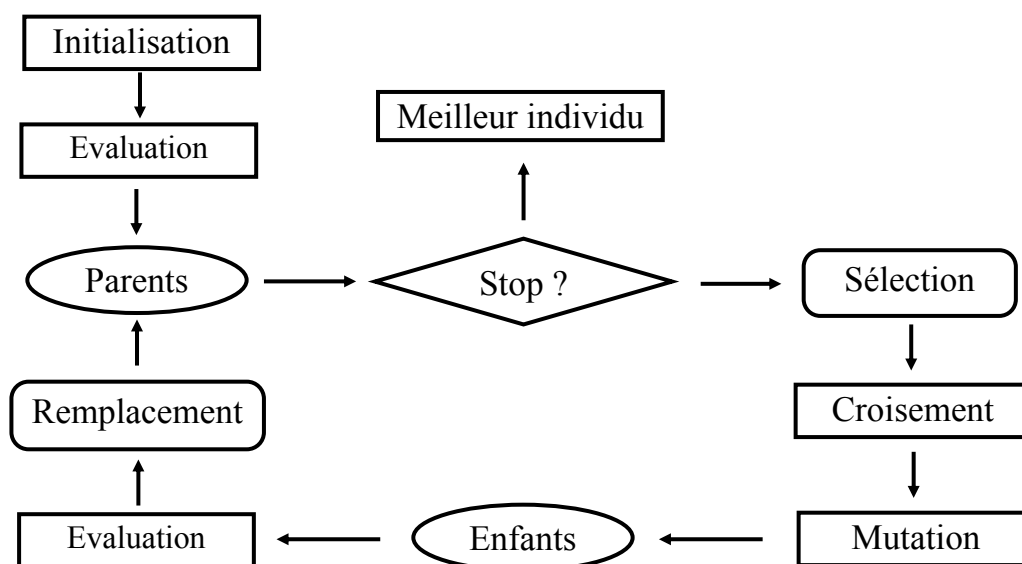


FIGURE 2.2: Le cycle d'un algorithme évolutionnaire

2.2.2.1 Codage de l'information

Le codage d'un individu est une représentation de ses caractéristiques, ou la définition d'une convention qui permet de décrire chaque individu sous la forme d'une chaîne de caractères.

Le codage doit être capable de tenir compte de toutes les caractéristiques principales du problème, cette représentation doit être aussi manipulable par les opérateurs génétiques tels que le croisement et la mutation. Aussi, pour chaque individu (solution possible) on doit définir une valeur de fitness qui reflète sa qualité [Ber10]. Il existe plusieurs types de codage des individus, les plus connus sont le codage binaire et le codage réel.

2.2.2.2 Principe de sélection

L'opération de sélection permet d'identifier les meilleurs individus d'une population, selon un ensemble de critères donnés, seuls les individus sélectionnés seront autorisés à se reproduire [Dur04]. On trouve dans la littérature plusieurs méthodes de sélection, chaque technique est mieux adaptée à une catégorie de problèmes. Les deux méthodes de sélection les plus répandues sont :

- **Sélection par roulette** : chaque individu de la population est représenté par une section dans une roulette, cette section est proportionnelle à sa fitness afin de favoriser les individus avec les meilleures fitness [Ber10]. À chaque lancement de la roulette on sélectionne un individu gagnant.
- **Sélection par tournoi** : cette méthode utilise la sélection par roulette sur des paires d'individus [Ter06], ensuite on sélectionne parmi ces paires, l'individu qui a la meilleure fitness.

2.2.2.3 Remplacement des individus

L'objectif de la procédure de remplacement consiste à réintroduire les nouveaux individus obtenus par les différents opérateurs génétiques, en remplacent leurs parents. Le rapport entre le nombre de descendants et le nombre d'individus parents est nommé generation gap [Gre86]. Il existe plusieurs stratégies de remplacement :

- Le remplacement systématique des plus mauvais individus.
- Le remplacement aléatoire.
- Le remplacement d'un pourcentage des individus de l'ancienne génération par les meilleurs enfants.

2.2.2.4 Opérateurs de variation

Les opérateurs de variation sont un ensemble des opérations stochastiques, c'est la phase où les individus enfants seront créés à partir des individus parents sélectionnés. Il y a deux opérateurs principaux, l'opérateur de croisement et l'opérateur de mutation [CRP98].

- A. Le croisement** : Le croisement est un phénomène naturel, qui peut être résumé dans l'échange du matériel génétique entre les parents en espérant que quelques enfants recevront les meilleures caractéristiques de leurs parents [SP94a].

Le croisement peut aussi augmenter la diversité de la population en manipulant la structure des chromosomes, ce qui permet d'explorer l'ensemble des solutions possibles. On distingue plusieurs types des opérateurs de croisement :

- Croisement à un point : consiste à sélectionner une position de coupure aléatoirement dans les deux individus parents, afin de générer deux nouveaux individus en permutant les parties droites dans les deux parents à partir de point de coupure (Figure 2.3).
- Croisement multi-points : consiste à définir k points de coupure et à permuter ensuite les différentes parties comme pour le croisement à un point.

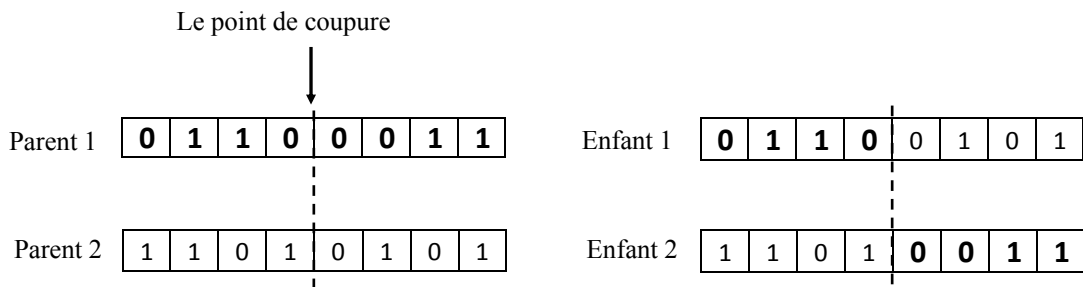


FIGURE 2.3: Le croisement

B. La mutation : Une modification aléatoire à un certain nombre des gènes dans un individu, afin d'introduire la diversité dans la population. La mutation est utilisée généralement avec un faible taux de probabilité (Figure 2.4) [SP94b].

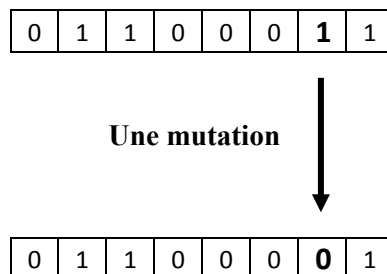


FIGURE 2.4: La mutation

2.3 Principes des algorithmes évolutionnaires quantiques

Les algorithmes évolutionnaires inspirés du quantique représentent le résultat de la combinaison entre les algorithmes évolutionnaires et le calcul quantique. Les recherches dans ce domaine ont commencé durant les années 90 [SBB⁺99] selon deux axes principaux, la

première approche basée sur le calcul purement quantique, tandis que la deuxième approche utilise les algorithmes évolutionnaires classiques avec les concepts de l'informatique quantique tels que les qubits, la superposition d'états, l'enchevêtrement et l'observation quantique.

Les AEQs ont été présentés pour la première fois par Narayanan et Moore [NM96], afin de résoudre le problème du voyageur de commerce, ensuite Han et Kim [HK00] ont proposé un modèle basé sur l'utilisation des qubits avec l'observation probabiliste, la superposition et l'utilisation de la porte quantique d'interférence, ce modèle a été appliqué sur le problème de Knapsack d'optimisation combinatoire.

Plusieurs algorithmes ont vu le jour ces dernières années, et les résultats obtenus suite à l'utilisation des algorithmes évolutionnaires quantiques sont très prometteurs.

2.3.1 Représentation quantique dans les AEQs

Les individus dans les AEQs sont fondés sur une codification quantique, un chromosome est représenté par un registre quantique ou une chaîne de qubits, cette représentation probabiliste est basée sur les concepts de la superposition et l'observation des qubits. En informatique quantique, un qubit est défini comme suit :

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad (2.1)$$

Selon le principe de superposition, le qubit est dans l'état 0 avec une probabilité de $|\alpha|^2$, et dans l'état 1 avec une probabilité de $|\beta|^2$, où $|\alpha|^2 + |\beta|^2 = 1$.

Un chromosome tel que décrit par Han et Kim [HK00] est un registre quantique de m qubits :

$$\begin{pmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_m \\ \beta_1 & \beta_2 & \cdots & \beta_m \end{pmatrix} \quad (2.2)$$

Cette définition de l'individu ne représente pas qu'une seule solution, mais une représentation de toutes les solutions possibles dans l'espace de recherche, c'est la puissance de la définition probabiliste du qubit. Théoriquement, un seul individu est suffisant pour couvrir complètement l'espace de recherche d'un problème donné.

Nous prenons ici un exemple d'un registre quantique à trois qubits [HK00] :

$$\begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{2} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & \frac{\sqrt{1}}{\sqrt{2}} \end{pmatrix}$$

Avec la représentation classique, on peut avoir une solution par individu, mais avec cette représentation quantique on peut avoir ($2^3 = 8$) solutions possibles par individu, chaque solution avec une probabilité différente.

Le système peut être représenté comme suit :

$$\frac{1}{4} |000\rangle + \frac{\sqrt{3}}{4} |001\rangle - \frac{1}{4} |010\rangle - \frac{\sqrt{3}}{4} |011\rangle + \frac{1}{4} |100\rangle + \frac{\sqrt{3}}{4} |101\rangle - \frac{1}{4} |110\rangle - \frac{\sqrt{3}}{4} |111\rangle$$

La probabilité pour chaque état dans le système est :

$$\begin{aligned} |000\rangle &= \frac{1}{16}, & |001\rangle &= \frac{3}{16}, & |010\rangle &= \frac{1}{16}, & |011\rangle &= \frac{3}{16} \\ |100\rangle &= \frac{1}{16}, & |101\rangle &= \frac{3}{16}, & |110\rangle &= \frac{1}{16}, & |111\rangle &= \frac{3}{16} \end{aligned}$$

Soit :

$$\begin{aligned} |000\rangle &= 6.25\%, & |001\rangle &= 18.75\%, & |010\rangle &= 6.25\%, & |011\rangle &= 18.75\% \\ |100\rangle &= 6.25\%, & |101\rangle &= 18.75\%, & |110\rangle &= 6.25\%, & |111\rangle &= 18.75\% \end{aligned}$$

A partir de cet exemple nous pouvons constater qu'un seul chromosome quantique de 3 qubits peut couvrir le même espace de solutions qu'une population de 8 chromosomes classiques.

2.3.2 Observation dans les AEQs

L'objectif de l'opération de mesure (observation) est d'obtenir un individu binaire à partir d'un individu quantique, elle consiste à une évaluation probabiliste des valeurs de $|\alpha|^2$ et $|\beta|^2$ pour chaque qubit du chromosome, par exemple :

$$\begin{pmatrix} 0.9877 & 0.4540 & 0.9998 & 0.0973 \\ 0.1564 & 0.8910 & 0.0175 & 0.9501 \end{pmatrix} \rightarrow \begin{pmatrix} 0 & 1 & 0 & 1 \end{pmatrix}$$

L'opération de mesure dans les AEQs ne modifie pas la valeur de qubit contrairement aux algorithmes quantiques sur un vrai ordinateur quantique. Un des modèles d'observations les plus répandus est celui de Han et Kim [HK00], la méthode d'observation binaire de

Han et Kim est comme suit :

```

début
|  si  $random(0,1) < |\alpha|^2$  alors
|  |  $x \leftarrow 0$ 
|  sinon
|  |  $x \leftarrow 1$ 
|  fin
fin
    
```

Algorithme 1 : Observation binaire (Han et Kim)

2.3.3 Opérations de mise à jour

Plusieurs opérateurs de variation sont utilisés dans les AEQs, afin de diversifier l'ensemble des solutions, ce qui permet de trouver de meilleures solutions, cette variation est réalisée en modifiant les valeurs de α et β . Notons bien que la condition $|\alpha|^2 + |\beta|^2 = 1$ doit être toujours satisfaite [Mah06].

Le choix de la porte quantique à utiliser est fait suivant la nature du problème à traiter [Mah06], parmi les portes quantiques on a par exemple : la porte Not, la porte Controled-Not, la porte Hadamard et la porte de rotation.

2.3.3.1 Porte de rotation quantique

La porte de rotation (interférence) quantique est la porte de mise à jour la plus utilisée dans les AEQs, cette porte effectue une rotation à l'état de qubit, dont le but d'augmenter la probabilité que le résultat de la prochaine opération de mesure correspond à la valeur de la meilleure solution actuelle.

$$\begin{pmatrix} \alpha' \\ \beta' \end{pmatrix} = U(\theta) \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad (2.3)$$

Où α' et β' représentent le qubit produit de la rotation $U(\theta)$ avec le qubit de la génération précédente, $U(\theta)$ peut être représentée sous la forme matricielle suivante :

$$U(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \quad (2.4)$$

Où θ représente l'angle de rotation. La valeur de θ détermine la vitesse de la convergence de l'algorithme, alors que le signe détermine le sens de recherche dans l'espace de recherche. Il faut bien noter qu'une grande valeur de θ peut guider l'algorithme vers un optimum local, tandis qu'avec une petite valeur de θ l'évolution de l'algorithme peut devenir très lente.

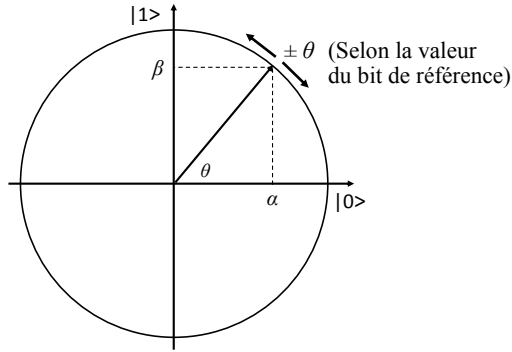


FIGURE 2.5: La porte de rotation quantique

La relation entre l'angle et la direction de rotation par rapport aux valeurs de α et β et le bit de référence [TBD04] est donnée dans le tableau 2.1 :

α	β	Valeur du bit de référence	Angle de rotation
> 0	> 0	1	$+\theta$
> 0	> 0	0	$-\theta$
> 0	< 0	1	$-\theta$
> 0	< 0	0	$+\theta$
< 0	> 0	1	$-\theta$
< 0	> 0	0	$+\theta$
< 0	< 0	1	$+\theta$
< 0	< 0	0	$-\theta$

Tableau 2.1: L'angle et la direction de rotation

2.3.3.2 Croisement quantique

D'une manière similaire aux algorithmes évolutionnaires classiques, l'opérateur de croisement quantique prend deux chromosomes parents et le résultat est deux chromosomes enfants, qui ont quelques caractéristiques des deux parents. L'avantage de l'utilisation du croisement quantique est qu'on peut avoir plus de diversité par rapport à l'opérateur de rotation (Figure 2.6)[TBD04].

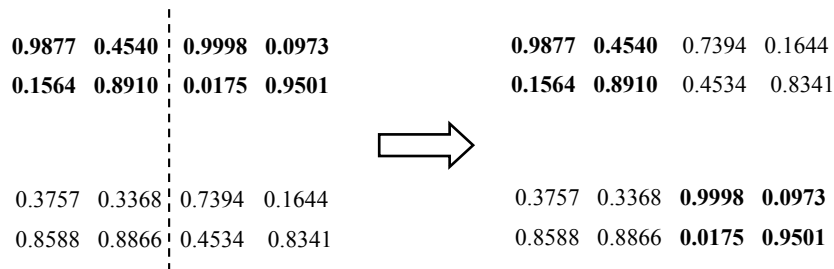


FIGURE 2.6: Le croisement quantique

2.3.3.3 Mutation quantique

La mutation quantique peut être définie par l'application d'un opérateur de négation quantique sur un certain nombre de qubits d'un individu, et en conséquence le changement de la probabilité de mesure de ce qubit (Figure 2.7)[Tal09].

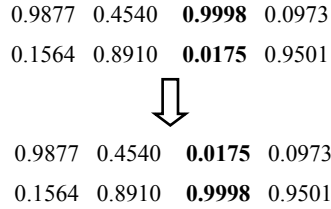


FIGURE 2.7: La mutation quantique

2.3.3.4 Permutation quantique

La permutation quantique peut être résumée à un changement d'emplacement entre deux qubits, elle peut jouer un rôle important dans la recherche de la solution optimale (Figure 2.8) [Tal09].

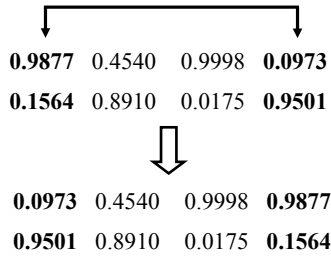


FIGURE 2.8: La permutation quantique

2.3.4 Structure générale des AEQs

Un algorithme évolutionnaire quantique tel que décrit par Han et Kim en 2000 [HK00] est composé des étapes suivantes (Algorithme 2) :

- i). **L'initialisation** : c'est l'étape de création de la population quantique initiale de n individus.

$$Q(t) = \{q_1^t, q_2^t, \dots, q_n^t\} \quad (2.5)$$

Où t représente la génération actuelle. Chaque individu dans la population contient m qubits.

$$q_j^t = \begin{pmatrix} \alpha_1^t & \alpha_2^t & \dots & \alpha_m^t \\ \beta_1^t & \beta_2^t & \dots & \beta_m^t \end{pmatrix}, \quad j = 1, 2, \dots, n \quad (2.6)$$

Pour la première génération $t = 0$, les valeurs de α_i^t et β_i^t pour $i = 1, 2, \dots, m$ de l'entière population sont initialisés par la valeur $\frac{1}{\sqrt{2}}$, donc la population $Q(0)$ est

dans une superposition de toutes les états possibles avec la même probabilité.

$$|\psi_{q_j^0}\rangle = \sum_{k=0}^{2^m} \frac{1}{\sqrt{2^m}} |S_k\rangle \quad (2.7)$$

Où S_k est le k^{eme} état représenté par la chaîne binaire (x_1, x_2, \dots, x_m) tel que x_i peut prendre une des deux valeurs 0 ou 1.

- ii). **L'observation** : c'est l'étape de mesure comme décrite dans la section précédente, c'est une opération probabiliste où le résultat est une population binaire $P(t) = \{X_1^t, X_2^t, \dots, X_n^t\}$ obtenue à partir des individus de la population quantique $Q(t)$.
- iii). **L'évaluation** : dans cette étape, on associe à chaque individu binaire (solution) X_j^t de la population binaire $P(t)$, une valeur de fitness qui représente sa qualité.
- iv). **La sélection** : les meilleures solutions seront stockées afin de les utiliser ensuite dans les opérations de mise à jour.
- v). Similaire à l'étape (ii).
- vi). Similaire à l'étape (iii).
- vii). **La mise à jour** : les opérations de mise à jour $U(t)$ sont un ensemble d'opérations basées sur les différentes portes quantiques, utilisées pour diriger les individus de la population vers les meilleures solutions.
- viii). Similaire à l'étape (iv).

début

$t \leftarrow 0;$

(i) Initialiser $Q(t)$;

(ii) Produire $P(t)$ par l'observation de $Q(t)$;

(iii) Evaluer $P(t)$;

(iv) Stocker la meilleure solution de $P(t)$;

tant que critère d'arrêt non satisfait faire

$t \leftarrow t + 1;$

(v) Produire $P(t)$ par l'observation de $Q(t-1)$;

(vi) Evaluer $P(t)$;

(vii) Mettre à jour $Q(t)$ en appliquant $U(t)$;

(viii) Stocker la meilleure solution de $P(t)$;

fin

fin

Algorithme 2 : La structure générale d'un AEQ

2.3.5 Types des AEQs

Dans le début des années 2000, après le succès de l'algorithme de Han et Kim [HK00], plusieurs variantes de ce type d'algorithmes sont apparues, de ce fait la classification et la mesure de performance de chaque type de ces algorithmes est devenue nécessaire. Les AEQs peuvent être classés selon plusieurs critères tels que le type d'observation utilisée (binaire, réel), ou les opérations de mise à jour appliquées par l'algorithme (rotation, croisement, mutation). D'après Zhang [Zha11] les algorithmes évolutionnaires quantiques peuvent être classés comme suit (Figure 2.9) [Ram13] :

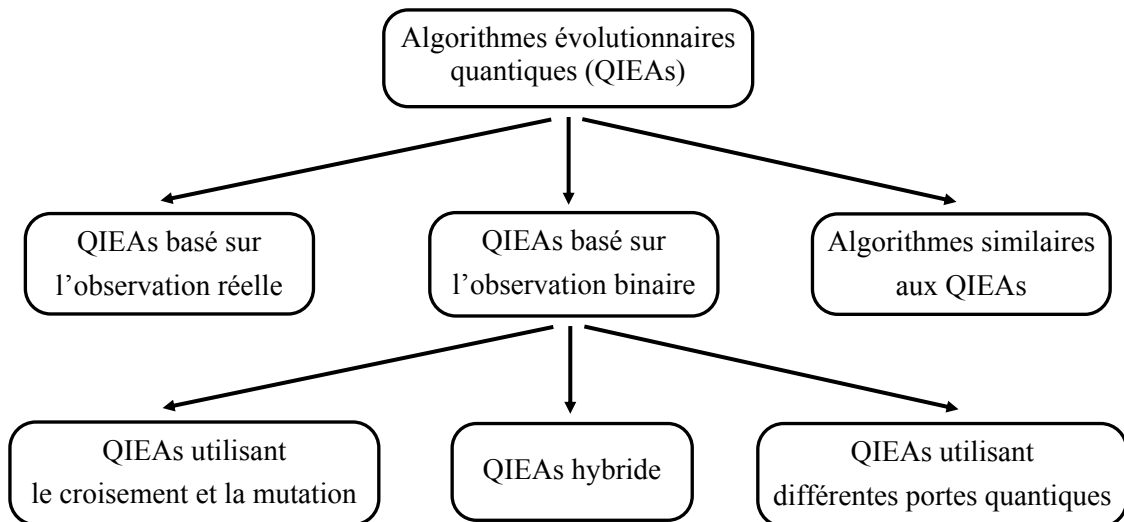


FIGURE 2.9: Les types des AEQs

2.3.5.1 AEQs basés sur l'observation binaire (bAEQ)

Parmi les premiers algorithmes évolutionnaires quantiques, proposés par Han et Kim en 2000 [HK00]. Les bAEQs sont basés sur une opération de mesure probabiliste binaire, et pour la mise à jour des chromosomes ils utilisent seulement la porte de rotation, d'après Han et Kim [HK00] les opérations de croisement et de mutation ont un effet négatif sur la convergence de l'algorithme.

2.3.5.2 bAEQs avec des opérations de croisement et de mutation (bAEQcm)

La version modifiée de l'algorithme de Han et Kim, cette méthode utilise le même processus d'observation binaire [Zha11], mais la porte de rotation est remplacée par les opérations de croisement et de mutation quantique pour la mise à jour des chromosomes.

2.3.5.3 AEQs basés sur l'observation réelle (rAEQ)

Les rAEQs sont basés sur le processus de mesure probabiliste réel de Zhang et Rong [ZR07] (Algorithme 3), et la porte de rotation quantique pour la mise à jour des individus

quantiques, les rAEQs ont été utilisés dans la résolution des problèmes d'optimisations numériques globales avec des variables continues.

```

début
|   si  $random(0,1) < 0.5$  alors
|   |  $x \leftarrow |\alpha|^2$ 
|   sinon
|   |  $x \leftarrow |\beta|^2$ 
|   fin
fin

```

Algorithme 3 : Observation réelle (Zhang et Rong)

Il existe d'autres variantes des algorithmes évolutionnaires quantiques, chaque'un de ces algorithmes utilise différentes techniques de mise à jour et de mesure, ces algorithmes peuvent être trouvés dans [Zha11].

Une étude comparative a été faite par Zhang [Zha11] entre un algorithme génétique classique et plusieurs variantes des algorithmes évolutionnaires quantiques, l'étude a été réalisée sur le problème de Knapsack (sac à dos) qui est un problème d'optimisation combinatoire [MT90], dans le but de comparer la performance des différents algorithmes.

Les résultats obtenus par ces tests ont montré l'efficacité des algorithmes évolutionnaires quantiques par rapport aux algorithmes génétiques classiques, et que la variante bAEQo à obtenue le meilleur résultat suivi par bAEQcm ensuite bAEQn et enfin l'algorithme génétique classique. Dans une autre comparaison entre les AEQs réels et binaires, les résultats illustrent que la variante rAEQ a donné de meilleurs résultats que les variantes binaires bAEQo-H et bAEQo-R.

D'autres tests réalisés par Zhang [Zha11] ont montré la supériorité des algorithmes évolutionnaires quantiques avec toutes ses variantes par rapport aux algorithmes génétiques classiques.

2.3.6 Caractéristiques des AEQs

Les algorithmes évolutionnaires quantiques sont caractérisés par les points suivants [Ram13] :

- La représentation dans les AEQs est basée sur les concepts de l'informatique quantique, la population est composée d'un ensemble d'individus ou chromosomes.
- Chaque individu est représenté par un registre quantique, cette représentation quantique est basée sur le principe de superposition d'états et l'observation probabiliste binaire ou réelle.
- La puissance d'un algorithme évolutionnaire quantique réside dans le fait qu'un petit nombre d'individus permet de couvrir l'espace de recherche, théoriquement

avec un seul individu.

- La mise à jour des individus dans les AEQs est réalisée à l'aide d'un ensemble de portes quantiques, afin de diriger la population vers les meilleures solutions, ainsi que la bonne exploration de l'espace de recherche.

2.3.7 Renforcement et Diversification dans les AEQs

Un algorithme génétique est un processus itératif dans le but est de trouver la bonne solution dans un espace de recherche, la convergence de l'algorithme peut être atteinte en se basant sur deux principes, la diversification et le renforcement [Tal09].

- La diversification cherche à explorer tout l'espace de recherche, pour trouver des nouvelles solutions qui peuvent être ensuite des solutions optimales.
- Le renforcement permet de raffiner la recherche, en explorant les zones proches aux meilleures solutions actuelles dans l'espace de recherche.

Pour que l'algorithme trouve la solution optimale, un bon équilibre entre les deux phénomènes doit être respecté, le résultat d'un taux de diversification élevé est une recherche aléatoire, alors qu'avec un taux de renforcement élevé on ne peut pas explorer suffisamment l'espace de recherche, et en conséquence on risque de ne pas atteindre les zones des meilleures solutions.

Dans les algorithmes évolutionnaires quantiques le renforcement est assuré par l'opérateur de rotation quantique [TBD04], qui permet à chaque itération d'augmenter la probabilité que la prochaine opération de mesure résultera une valeur similaire à celle de la meilleure solution actuelle.

Le problème si on ne trouve pas une nouvelle meilleure solution pour beaucoup d'itérations, alors la probabilité sera très élevée que la valeur inverse ne se produira jamais, ce qu'on appelle le problème d'optimum local.

Notons bien que le bon choix de l'angle de rotation à un grand effet, une valeur élevée peut donner un optimum local, tandis que une petite valeur résultera une vitesse de convergence très lente.

En revanche, la diversification peut être assurée par les opérateurs du croisement et de mutation quantique, qui peuvent être utilisés pour explorer tout l'espace de recherche, et remédier le problème d'optimum local engendré par l'opérateur de rotation.

2.3.8 Avantages des AEQs

Les algorithmes évolutionnaires quantiques présentent plusieurs avantages, parmi ces avantages nous citons :

- Le principe de la superposition permet de réduire énormément la taille de la population à quelques individus.
- La vitesse de convergence de l'algorithme peut être contrôlée par l'angle de rotation.
- Les opérateurs quantiques utilisés par les AEQs permettent une balance automatique entre la recherche locale et la recherche globale.
- Les AEQs permettent un traitement massivement parallèle.
- Avec les AEQs, l'historique des itérations successives sur un individu n'est pas perdu grâce à sa nature probabiliste.

2.4 Conclusion

Dans ce chapitre, nous avons présenté les algorithmes évolutionnaires inspirés du quantique où nous avons montré comment les concepts de l'informatique quantique comme la superposition d'états, l'observation probabiliste et les portes quantiques sont appliqués, et comment ils ont amélioré la performance des algorithmes évolutionnaires classiques.

Nous avons exposé aussi les différentes techniques d'observation probabiliste, les opérations quantiques de mise à jour, ainsi que les divers types d'algorithmes évolutionnaires quantiques et quelques tests de performance entre ces algorithmes, et enfin nous avons discuté les phénomènes de renforcement et de diversification et leurs effets au comportement de l'algorithme à l'exploration de l'espace de recherche.

Depuis l'algorithme original de Han et Kim en 2000, les algorithmes évolutionnaires quantiques sont devenus populaires, ils ont été utilisés pour résoudre beaucoup de problèmes où ils ont montré de bons résultats tout en utilisant une population plus petite et moins d'itérations, ce qui donne beaucoup d'espérances pour cette technique prometteuse.

Descripteurs et mesures de similarité

Ce chapitre est consacré à la présentation de l'appariement de formes. Nous exposons les différentes techniques de représentation de la forme, ainsi que les mesures de similarité entre les formes, et quelques modèles de transformations géométriques.

3.1	Introduction	47
3.2	Quelques définitions	47
3.3	Paramètres de forme	48
3.3.1	Centre de gravité	48
3.3.2	Axe de moindre inertie	48
3.3.3	Énergie de pliage	49
3.3.4	Excentricité	49
3.3.5	Rapport de circularité	49
3.3.6	Rectangularité	50
3.3.7	Convexité	50
3.3.8	Solidité	50
3.3.9	Nombre d'Euler	50
3.3.10	Profils	51
3.3.11	Rapport de la surface du trou	51
3.3.12	Invariance aux transformations géométriques	51
3.4	Descripteurs de formes	52
3.4.1	Descripteurs basés région	55
3.4.2	Descripteurs basés contour	57
3.4.3	Ensemble fini de points	59
3.4.4	L'invariance et la résistance des descripteurs de forme	60
3.5	Mesures de similarité	60
3.5.1	Distance métrique	61
3.5.2	Distance non métrique	64
3.6	Transformations	66

3.6.1	Translation	66
3.6.2	Transformation rigide	66
3.6.3	Similitude	66
3.6.4	Transformation affine	67
3.6.5	Transformation projective	67
3.7	Conclusion	67

3.1 Introduction

L'importance de l'information visuelle est très élevée actuellement dans plusieurs domaines comme la médecine, le journalisme et la publicité, nous trouvons des milliers d'images stockées, l'exploitation de ces images d'une manière efficace représente un des défis de l'informatique.

L'objectif principal des recherches dans le domaine de la vision par ordinateur est le développement des techniques permettant le traitement et la reconnaissance automatique des objets dans les images et les vidéos. L'appariement de forme représente un sujet très important en vision par ordinateur, il est utilisé pour résoudre plusieurs problèmes comme la reconnaissance d'objets et la recherche d'images par le contenu.

Dans ce chapitre, nous allons présenter les différents concepts et techniques utilisés dans l'appariement de formes, comme les techniques de représentation de formes et les mesures de similarité entre les formes.

3.2 Quelques définitions

L'appariement de formes : Les bases d'images sont utilisées dans plusieurs domaines aujourd'hui, chaque une de ces bases d'images contient un nombre très élevé d'images. La recherche classique par les descripteurs extérieurs, comme le texte associé, est devenue insuffisante [VH01]. En revanche, l'utilisation des descripteurs basés sur le contenu visuel de l'image a attiré plus d'attention les dernières années.

L'appariement de formes (shape matching) est parmi les plus importants sujets dans le domaine de la vision par ordinateur, il représente un élément essentiel dans la recherche d'images basée sur la forme géométrique. L'appariement de formes traite le problème de la mesure de similarité entre deux formes selon plusieurs techniques de mesure, ainsi que le problème de la transformation entre ces deux formes [VH00].

L'appariement de formes peut être utilisé dans d'autres applications en vision par ordinateur comme le recalage d'images, la reconnaissance d'objets, etc.

Recalage d'images : (image registration) c'est une technique qui consiste de trouver la transformation optimale permettant de superposer la plus grande partie commune entre deux images [Bro92].

Reconnaissance d'objets : (object recognition) en vision par ordinateur, la reconnaissance d'objets est une technique qui consiste d'identifier des classes d'objets dans des images ou vidéos [HAR11].

La recherche d'images par le contenu : c'est une technique qui permet de trouver les images à partir de leurs caractéristiques visuelles, il existe deux approches principales, la première basée sur l'intensité (couleur et texture), alors que la deuxième est basée sur la forme géométrique d'objets contenus dans l'image [Yan08] ; les études ont montré que les utilisateurs sont plus intéressés par les techniques basées sur la forme que par celles basées sur l'intensité [SdLV99].

L'approximation de formes : construire une forme similaire à l'originale, mais avec moins d'éléments (points, segments, etc.) [YKR⁺08].

3.3 Paramètres de forme

Certaines caractéristiques géométriques simples peuvent être employées à la mesure de similarité entre les formes, ces caractéristiques ne peuvent généralement discriminer que les formes clairement dissimilaires où la différence est importante. Pour cette raison, elles sont généralement utilisées comme des filtres pour éliminer les faux résultats ou combinées avec d'autres descripteurs de forme [Yan08], elles ne sont pas capables d'être les seuls descripteurs de forme. Une forme peut être décrite par différents paramètres, parmi ceci, nous citons :

3.3.1 Centre de gravité

Le centre de gravité (centroïde) a une position fixe par rapport à la forme, pour une forme représentée par sa région, le centre de gravité (g_x, g_y) est donné par :

$$g_x = \frac{1}{N} \sum_{i=1}^N x_i \quad (3.1)$$

$$g_y = \frac{1}{N} \sum_{i=1}^N y_i \quad (3.2)$$

Où N représente le nombre de points de la forme et $(x_i, y_i) \in \{(x_i, y_i) | f(x_i, y_i) = 1\}$

3.3.2 Axe de moindre inertie

L'axe de moindre inertie (Axis of least inertia) est unique à la forme, il sert comme une ligne de référence unique pour préserver l'orientation de la forme. L'axe de moindre inertie d'une forme est défini comme la ligne pour laquelle l'intégrale de carré des distances aux points sur le contour de la forme est un minimum [Yan08]. L'axe de moindre inertie passe généralement à travers le centre de gravité de la forme.

3.3.3 Énergie de pliage

L'énergie de pliage moyenne (Average bending energy) est définie par :

$$BE = \frac{1}{N} \sum_{s=0}^{N-1} K(s)^2 \quad (3.3)$$

Où $K(s)$ est la fonction de courbure, (s) est le paramètre de longueur d'arc et N le nombre de point du contour [Lon98].

3.3.4 Excentricité

L'excentricité (eccentricity) est la mesure du rapport d'aspect. C'est le rapport de la longueur du grand axe par la longueur de petit axe, elle peut être calculée par la méthode des axes principaux (principal axes) (Figure 3.1) ou par la méthode de rectangle à limite minimum (minimum bounding rectangle) [PI97].

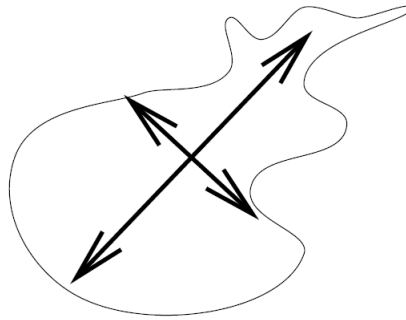


FIGURE 3.1: La méthode des axes principaux

3.3.5 Rapport de circularité

Le rapport de circularité (circularity ratio) représente la façon dont une forme est semblable à un cercle. Il est défini comme le rapport entre la surface de la forme et la surface d'un cercle ayant le même périmètre.

$$Circularité = \frac{A_S}{A_C} \quad (3.4)$$

Où A_S est la surface de la forme, et A_C la surface d'un cercle ayant le même périmètre que la forme.

3.3.6 Rectangularité

La rectangularité représente la façon dont une forme est rectangulaire.

$$\text{Rectangularité} = \frac{A_S}{A_R} \quad (3.5)$$

Où A_S est la surface de la forme, et A_R est la surface de rectangle à limite minimum (minimum bounding rectangle).

3.3.7 Convexité

La convexité est définie comme le rapport entre le périmètre de l'enveloppe convexe (convex hull) (Figure 3.2) $O_{\text{Enveloppeconvexe}}$ et le contour original O [PI97] :

$$\text{Convexité} = \frac{O_{\text{Enveloppeconvexe}}}{O} \quad (3.6)$$

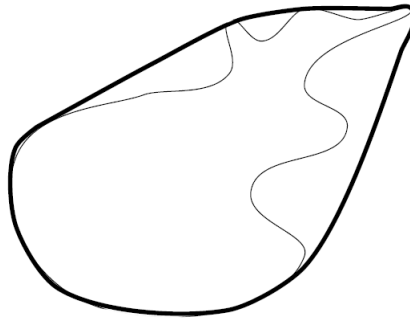


FIGURE 3.2: Une enveloppe convexe

3.3.8 Solidité

La solidité décrit à quel point la forme est convexe ou concave, la solidité d'une forme convexe est toujours 1, elle est définie par :

$$\text{Solidité} = \frac{A_S}{H} \quad (3.7)$$

Où A_S est la surface de la forme, et H représente la surface de l'enveloppe englobant de la forme.

3.3.9 Nombre d'Euler

Le nombre d'Euler est la différence entre le nombre des parties contiguës et le nombre de trous dans une forme. Soit S le nombre des parties contiguës et N le nombre de trous

dans une forme, le nombre d'Euler est [Yan08] :

$$\text{Nombre d'Euler} = S - N \quad (3.8)$$

Exemple :

3 B 9

Nombre d'Euler égal à 1, -1 et 0, respectivement.

3.3.10 Profils

Les profils représentent la projection d'une forme sur l'axe x et l'axe y dans le système de coordonnées cartésiennes, pour obtenir deux fonctions unidimensionnelles.

$$Pro_x(i) = \sum_{j=j_{min}}^{j_{max}} f(i, j) \quad (3.9)$$

$$Pro_y(j) = \sum_{i=i_{min}}^{i_{max}} f(i, j) \quad (3.10)$$

3.3.11 Rapport de la surface du trou

Le rapport de la surface du trou (Hole area ratio) est défini comme suit :

$$\text{Rapport du trou} = \frac{A_h}{A_S} \quad (3.11)$$

Où A_S est la surface de forme, et A_h la surface du trou.

3.3.12 Invariance aux transformations géométriques

L'invariance des paramètres de forme contre les différents types de transformations géométriques est résumée dans le tableau 3.1.

Caractéristiques géométriques	Translation	Mise à l'échelle	Rotation
L'énergie de pliage moyenne	Oui	Non	Oui
Excentricité	Oui	Oui	Oui
Rapport de circularité	Oui	Oui	Oui
Nombre d'Euler	Oui	Oui	Oui
Rectangularité	Oui	Oui	Oui
Convexité	Oui	Oui	Oui
Solidité	Oui	Oui	Oui
Profils	Oui	Oui	Non
Rapport de la surface du trou	Oui	Oui	Oui

Tableau 3.1: L'invariance des paramètres de la forme

3.4 Descripteurs de formes

La forme d'un objet physique est la géométrie de ses surfaces extérieures ou son contour externe, les frontières entre l'intérieur et l'extérieur de l'objet. La forme est le contour ou la surface d'un objet [Yan08]. Une forme peut être représentée par son contour, région ou un ensemble fini de points :

- Un contour peut être défini comme les frontières de l'objet, il s'agit d'une courbe continue sur un plan. Dans ce type de représentation, les frontières d'objet sont la seule information de la forme exploitée.
- Dans le cas de la représentation par région, tous les pixels de la forme sont pris en compte, une forme peut être composée par une seule région ou un ensemble des régions, ainsi que des trous dans l'objet.
- La représentation par un ensemble fini de points peut être considérée comme des points distincts sur les frontières d'objet, ces points sont prélevés afin d'obtenir une représentation de la forme. Cette technique cible les points les plus importants dans la forme et les plus sensibles à la vision humaine.

Les descripteurs de forme basés contour utilisent uniquement l'information des frontières en ignorant le contenu intérieur de la forme. Exemples des descripteurs de forme basés sur le contour, nous avons le descripteur de Fourier [ZL⁺01], le descripteur d'ondelettes [MO99], etc. Comme ils sont calculés en utilisant uniquement les pixels des frontières, leurs complexités de calcul est généralement faible et leurs descripteurs sont compacts.

Le contour peut être considéré comme une courbe numérique, donc on a besoin d'appliquer plusieurs méthodes géométriques pour son analyse. Cependant, ces descripteurs ne peuvent pas représenter suffisamment la forme en se basent uniquement sur l'information des frontières, par exemple : le problème d'objets avec des trous, les objets partiellement occultés et les objets complexes composés de plusieurs régions séparées, ainsi que leurs faiblesses contre le bruit.

Les descripteurs de forme basés région exploitent à la fois les frontières et les pixels intérieurs de la forme. Exemples des descripteurs de forme basés région : le descripteur Moments géométriques [CA05] et Matrice de forme [TS89], etc. Ces descripteurs utilisent tous les pixels de la forme et en conséquence, ils peuvent décrire les différents types de formes efficacement dans un seul descripteur. Ils sont également robustes aux petites déformations sur les frontières d'objets. En plus, la segmentation de régions est beaucoup plus facile que les techniques d'analyses utilisées pour le contour [Yan08]. Toutefois, la taille d'un descripteur basé région est généralement plus grande et génère une complexité de calcul plus importante lors de l'extraction des caractéristiques et la mesure de similarité.

La représentation la plus naturelle d'une forme dans une image consiste à une représentation par une image binaire comme suit :

$$f(i, j) = 0, \quad si(i, j) \notin objet \quad (3.12)$$

$$= 1, \quad si(i, j) \in objet \quad (3.13)$$

Dans le cas de plusieurs objets, on utilise des étiquettes :

$$f(i, j) = k, \quad si(i, j) \in objet \quad k \quad (3.14)$$

La technique de représentation de forme basée sur un ensemble fini de points représente le résultat d'une opération d'échantillonnage appliquée à un contour d'objet, exemples de ce type des descripteurs : nous avons shape contexts [BMP00] et voting schemes [VH01], etc.

L'ordre des points n'est généralement pas important pour ces descripteurs, ils peuvent extraire les caractéristiques d'un objet avec plusieurs trous ou même un objet complexe avec plusieurs parties. L'idée clé de la mesure de similarité avec cette approche est de trouver les meilleures correspondances de points entre les formes.

La figure 3.3 montre une représentation de la forme basée région (droite), et une représentation basée contour (centre), et une représentation basée sur l'ensemble fini de points (gauche).

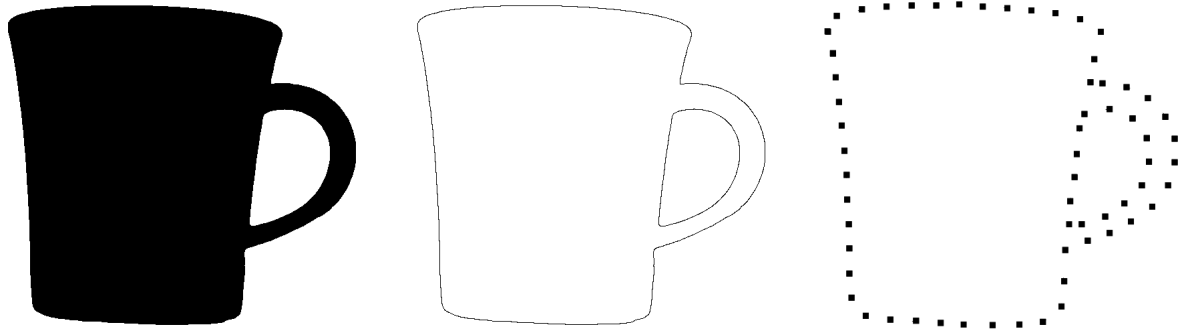


FIGURE 3.3: Les techniques de représentation de forme

Selon Zhang [ZL04], les méthodes de représentation de formes sont divisées en deux classes : basées contours et basées régions. Ces deux classes peuvent être divisées en deux sous-classes : approches structurelles et approches globales (Figure 3.4) [Ric05][ZL04].

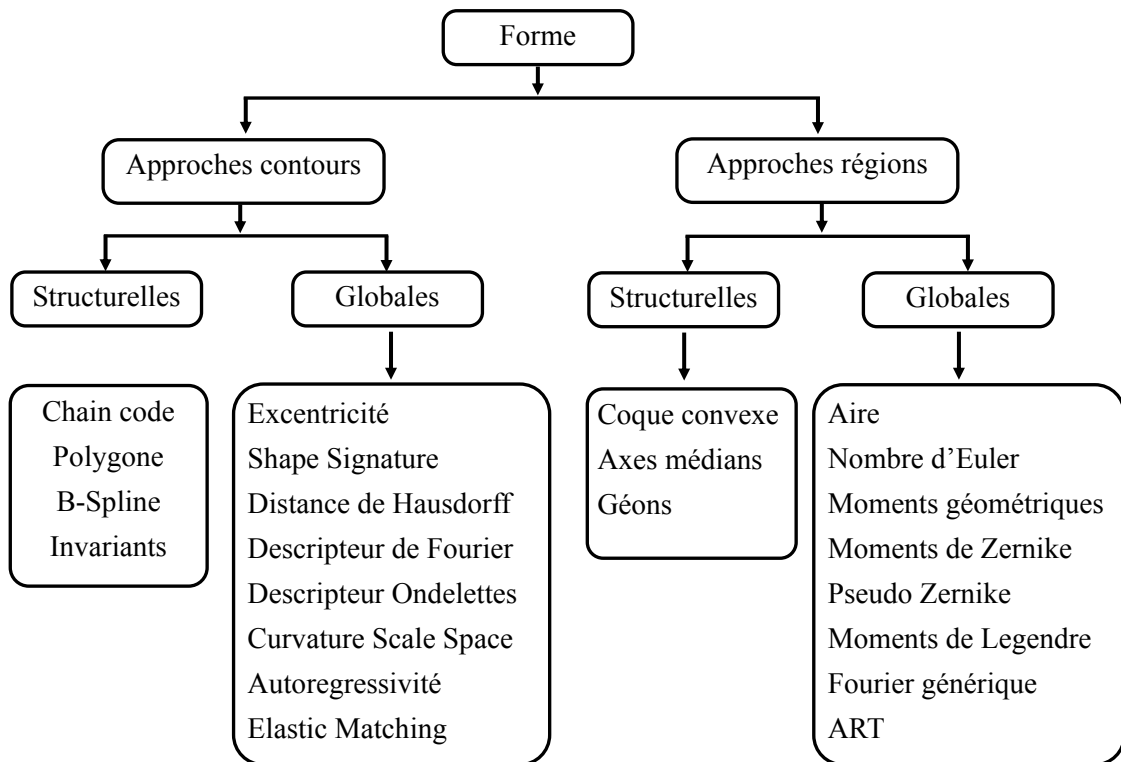


FIGURE 3.4: La classification des descripteurs de forme

Les approches structurelles décomposent la forme à des éléments de base appelés primitives selon un critère donné ; une phase de segmentation est réalisée dans le but d'extraire l'ensemble de primitives, ensuite la comparaison de ces primitives afin de mesurer la similarité entre les formes. La forme peut être décrite par les méthodes structurelles comme un graphe reliant les éléments de base, la comparaison est réalisée entre les éléments de graphe. Ces méthodes présentent deux problèmes majeurs : l'inexistence d'une définition formelle de la forme et l'incapacité de déterminer le nombre de primitives nécessaires. Aussi, la mesure de similarité dans les approches structurelles est non métrique. Un autre

problème est leurs faiblesses contre l'apparition de bruit qui limite les domaines de leurs utilisations [ZL04].

Les méthodes globales font une analyse complète de la forme, les descripteurs de cette technique représentent l'intégralité de la forme, et la mesure de similarité est appliquée directement sur les descripteurs, cette mesure de similarité est généralement métrique.

3.4.1 Descripteurs basés région

3.4.1.1 Moments

Un objet dans une image peut être décrit par un ensemble des moments $m_{p,g}$, le moment (p, g) d'un objet $O \subseteq R^2$ est défini par :

$$m_{p,g} = \int_{(x,y) \in O} x^p y^g dx dy \quad (3.15)$$

Pour un ensemble fini de points, l'intégral peut être remplacé par la somme :

$$m_{p,g} = \sum_x \sum_y x^p y^g f(x, y) \quad (3.16)$$

Les moments capturent l'information globale de l'image basée région, le moment d'ordre zéro m_{00} est égal à la surface de la forme en supposant que $f(x, y)$ est la fonction de silhouette. Les moments sont parmi les premiers descripteurs d'images dans la vision par ordinateur.

A partir de l'équation de moments, plusieurs fonctions peuvent être définies pour améliorer la définition classique des moments en rajoutant par exemple l'invariance à certaines transformations géométriques, par exemple Hu [Hu62] a décrit un ensemble de moments invariants à la translation et à la similitude.

3.4.1.2 Descripteur ART (Angular Radial Transform)

La technique ART est une transformation 2D complexe définie sur un disque unitaire [ZP04], la méthode ART est compacte et efficace dans la représentation des régions disjointes. La technique extrait un ensemble des coefficients ART f_{nm} d'ordre n et m , de chaque forme par la méthode suivante :

$$F_{nm} = Vnm(p, \theta), f(p, \theta) \quad (3.17)$$

$$= \sum_0^{2\pi} \sum_0^1 Vnm(p, \theta), f(p, \theta) p \quad (3.18)$$

Où $f(p, \theta)$ est la fonction d'intensité de l'image, et $Vnm(p, \theta)$ la base de la fonction définie sur le disque unitaire de façon séparable comme suit :

$$Vnm(p, \theta) = \frac{1}{2\pi} \exp(jm\theta) R_n(p), \quad (3.19)$$

$$R_n(p) = 1 \quad \text{si } n = 0, \quad (3.20)$$

$$= 2\cos(\pi np) \quad \text{si } n \neq 0 \quad (3.21)$$

3.4.1.3 Enveloppe convexe (Convex hull)

Une région R est dite convexe lorsque pour chaque deux points $x_1, x_2 \in R$, le segment $[x_1, x_2]$ qui les joint est entièrement contenu [ZL04]. L'enveloppe convexe est définie comme la plus petite région convexe H qui satisfait la condition $R \subset H$. La différence $H - R$ est appelée la carence convexe D de la région R .

3.4.1.4 Matrice de forme (Shape matrix)

La plus part des approches basées sur la matrice de forme place la forme dans une grille rectangulaire pour échantillonner l'information globale de la forme afin de créer une matrice binaire où chaque valeur dans la matrice représente un élément de grille ; les méthodes basées sur les grilles rectangulaires ne sont généralement pas invariantes à la translation, à la rotation et à la mise à l'échelle, alors la normalisation est nécessaire.

Goshtasby [Gos85] a utilisé le même principe d'échantillonnage mais il a remplacé la grille rectangulaire par une grille polaire de cercles concentriques et de lignes radiales positionnée au centre de masse de la forme (Figure 3.5). La matrice est formée de telle sorte que les cercles correspondent aux colonnes de la matrice et les lignes radiales correspondent aux lignes de la matrice. Cette méthode est invariante à la translation, à la rotation et à la mise à l'échelle.

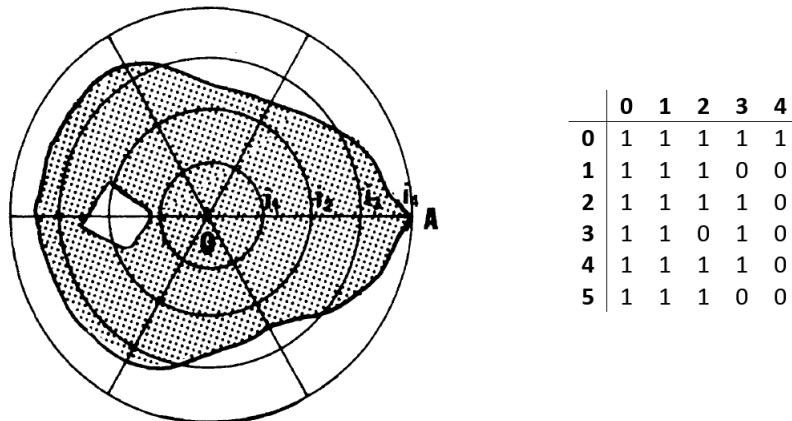


FIGURE 3.5: Une matrice de forme

3.4.1.5 Transformée d'axe médian

La transformée d'axe médian représente la forme par un graphe dont les caractéristiques importantes de la forme sont préservées [Ari03]. Le graphe est basé sur le squelette de la forme. La première étape est la construction du squelette, la manière la plus facile de squelettisation est l'amincissement d'une forme jusqu'à obtenir un ensemble des courbes centrées [Dav04]. Le squelette peut être défini comme un ensemble connexe de lignes médianes (Figure 3.6) [ZL04]. Un des points faibles de la méthode transformée d'axe médian est sa sensibilité au bruit du contour.

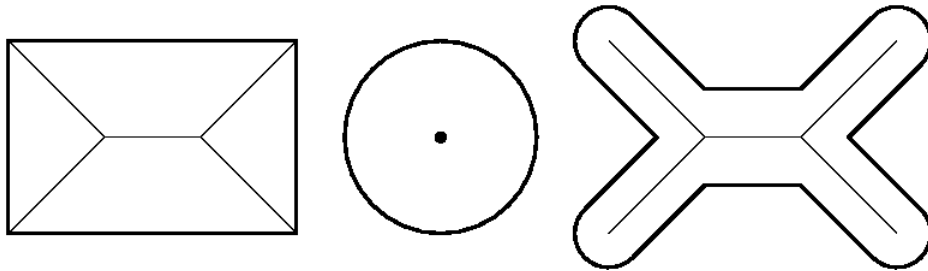


FIGURE 3.6: La construction du squelette

3.4.2 Descripteurs basés contour

3.4.2.1 Signatures de forme

Une signature de forme représente la forme par une fonction unidimensionnelle obtenue à partir des points de contour. Il existe plusieurs signatures de forme comme, l'angle de la tangente [ZL⁺01], courbure du contour [WLT99], etc. Les signatures de forme sont généralement normalisées afin d'être invariantes à la translation et à la mise à l'échelle. Cette technique à deux problèmes majeurs : le coût d'appariement élevé et la sensibilité contre le bruit.

3.4.2.2 Moments de frontière (Boundary moments)

Cette méthode peut être utilisée pour réduire les dimensions de la représentation du contour de la forme. Supposons que le contour de la forme est représenté par la signature de forme $z(i)$ alors le moment m_r et le moment central μ_r [SHB14] peuvent être définis comme suit :

$$m_r = \frac{1}{N} \sum_{i=0}^N [z(i)]^r \quad (3.22)$$

$$\mu_r = \frac{1}{N} \sum_{i=0}^N [z(i) - m_1]^r \quad (3.23)$$

Où N est le nombre de points du contour. L'avantage de cette technique qu'elle est facile à implémenter, les moments normalisés $\bar{m}_r = m_r/(\mu_2)^{r/2}$ et $\bar{\mu}_r = \mu_r/(\mu_2)^{r/2}$ sont invariants à la translation, à la rotation et à la mise à l'échelle. Pour avoir des descripteurs moins sensibles au bruit on peut utiliser :

$$F_1 = \frac{(\mu_2)^{1/2}}{m_1} \quad (3.24)$$

$$F_2 = \frac{\mu_3}{(\mu_2)^{3/2}} \quad (3.25)$$

$$F_3 = \frac{\mu_4}{(\mu_2)^2} \quad (3.26)$$

3.4.2.3 L'appariement élastique (Elastic matching)

Selon cette méthode, un modèle déformé peut être généré à partir de modèle d'origine $t(s)$ et de la déformation $\theta(s)$.

$$\varphi(s) = t(s) + \theta(s) \quad (3.27)$$

Où $t = (t_x, t_y)$ est une spline de deuxième ordre et $\theta = (\theta_x, \theta_y)$ représente la déformation [DBP97]. La similarité entre la forme originale du modèle et la forme de l'objet dans l'image est mesurée par la fonction suivante :

$$F = S + B + M \quad (3.28)$$

$$= x \int_0^1 \left[\left(\frac{d\theta_x}{ds} \right)^2 + \left(\frac{d\theta_y}{ds} \right)^2 \right] ds + \beta \int_0^1 \left[\left(\frac{d^2\theta_x}{ds^2} \right)^2 + \left(\frac{d^2\theta_y}{ds^2} \right)^2 \right] ds + \int_0^1 I_E(\varphi(s)) ds \quad (3.29)$$

Où I_E est l'image d'objet, S est l'énergie de déformation et B l'énergie de courbure et M la mesure de degré de chevauchement entre les deux formes.

3.4.2.4 Espace d'échelle (Scale space)

Le problème de la sensibilité au bruit et les variations du contour dans la plupart des méthodes du domaine spatial, dirige vers l'utilisation de la méthode de l'espace à échelle. La représentation en espace à l'échelle d'une forme est créée par le suivi des positions des points d'inflexions dans le contour d'une forme filtrée par des filtres passe-bas gaussien des largeurs variables. Chaque fois que la largeur de filtre gaussien augmente, les inflexions insignifiantes sont éliminées du contour et en conséquence la forme devient plus lisse et plus claire. Les points d'inflexion restant dans le contour sont les caractéristiques significatives de l'objet.

Une des variantes de la technique espace à l'échelle la plus utilisée est la méthode CSS (Curvature Scale Space), nous avons le contour C paramétré par la longueur d'arc s , $C(s) = (x(s), y(s))$ [Vel01], une convolution est appliquée sur la fonction de coordonnées

de C avec un noyau gaussien ϕ_σ de largeur σ :

$$x_\sigma(s) = \int x(s)\phi_\sigma(t-s)dt \quad (3.30)$$

$$y_\sigma(s) = \int y(s)\phi_\sigma(t-s)dt \quad (3.31)$$

$$\phi_\sigma(t) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{t^2}{2\sigma^2}} \quad (3.32)$$

Avec l'augmentation de la valeur de σ le contour obtenu devient plus lisse (Figure 3.7).

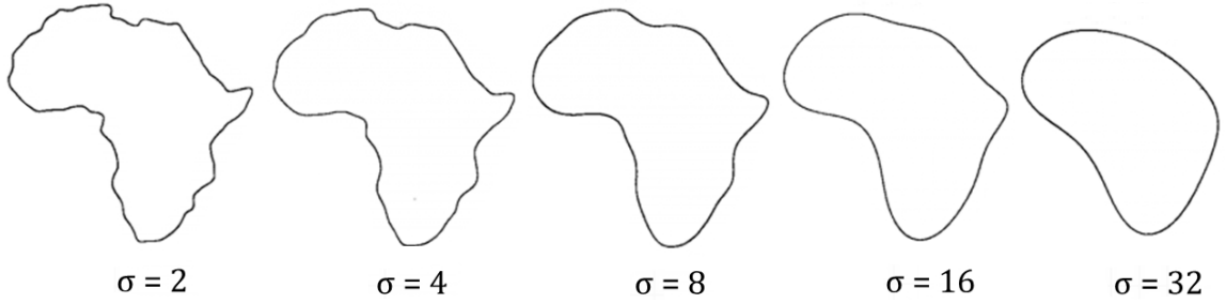


FIGURE 3.7: L'effet d'augmentation de la valeur de σ

3.4.2.5 Descripteur de Fourier

Le descripteur de Fourier décompose le contour de la forme en des fréquences obtenues par la transformée de Fourier, cette transformée est appliquée à la fonction du contour et les résultats sont des coefficients utilisés pour la représentation de la forme [Ari03]. La transformée de Fourier discrète pour une forme $u(s)$ est donnée comme suit :

$$a_n = \frac{1}{N} \sum_{s=0}^{N-1} u(s)exp(-j2\pi is/N) \quad (3.33)$$

Afin de réaliser l'invariance à la translation et à la rotation, l'information de phase de a_n est ignorée et nous n'utilisons que l'amplitude $|a_n|$. Les amplitudes sont divisées par $|a_0|$ pour obtenir l'invariance à la mise à l'échelle.

3.4.3 Ensemble fini de points

3.4.3.1 Contexte de forme (Shape context)

Cette méthode extrait une caractéristique globale de forme appelée shape context. La première étape consiste à la détection du contour et à la sélection d'un ensemble de n points, pour extraire les caractéristiques en un point p , des vecteurs de p vers tous les autres points sont tracés, la longueur r et l'orientation θ de ces vecteurs sont quantifiées pour créer un histogramme de distribution spatiale qui est utilisé pour représenter le point

p .

L'ensemble d'histogrammes de tous les points dans la forme représente le shape context. L'appariement entre deux formes est réalisé en trouvant l'ensemble des correspondances entre les points qui donne le coût d'appariement le plus faible [BMM06].

3.4.3.2 Contexte de forme à distance intérieure (Inner-distance shape contexts)

La méthode Inner-Distance Shape Contexts (IDSC) proposée par Ling [LJ07] pour remédier à la faiblesse de shape context à capturer les formes articulées. La distance intérieure (Inner-Distance) peut être définie comme la distance entre deux points de contour, dans l'intérieur du contour de la forme (Figure 3.8), ensuite la distance intérieure est utilisée pour remplacer la distance euclidienne pour construire l'histogramme de shape context.

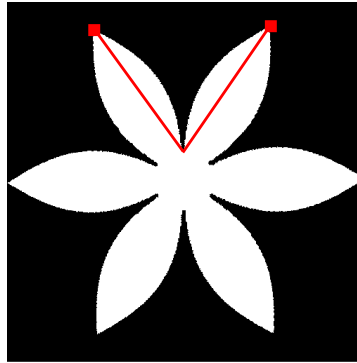


FIGURE 3.8: La distance intérieure entre deux points

3.4.4 L'invariance et la résistance des descripteurs de forme

L'invariance des descripteurs de forme contre les différents types de transformations géométriques, et leur résistance contre l'occultation et le bruit, ainsi que leur complexité de calcul est résumée dans le tableau 3.2 [Yan08].

3.5 Mesures de similarité

La mesure de similarité est parmi les éléments les plus importants dans l'appariement de formes, il existe deux directions de recherche pour ce problème : la première vise à obtenir les meilleurs résultats de recherche dans une base d'images en essayant de rendre les mesures de similarité aussi riches que possible, tandis que l'objectif de la deuxième est de développer des mesures de similarité plus efficaces afin de rendre la recherche dans les grandes bases d'images plus rapide.

Une mesure de similarité doit avoir les propriétés suivantes :

- Elle doit être universelle afin de nous permettre d'identifier des formes arbitraires.

Descripteur	Translation	l'échelle	Rotation	Bruit	Occultation	Complexité
Enveloppe convexe	Bon	Bon	Bon	Moyen	Mauvais	élevé
Matrice de forme carré	Bon	Bon	Bon	Mauvais	Moyen	Moyen
Matrice de forme polaire	Bon	Bon	Bon	Mauvais	Moyen	faible
Shape context	Bon	Bon	Bon	Mauvais	Moyen	Moyen
IDSC	Bon	Bon	Bon	Mauvais	Moyen	Moyen
ART	Bon	Bon	Bon	Bon	Mauvais	élevé
CSS	Bon	Bon	Bon	Bon	Bon	Moyen
Shock graphs	Bon	Bon	Bon	Bon	Bon	élevé
Moments de frontière	Bon	Bon	Bon	Moyen	Mauvais	faible
Moments algébrique	Bon	Bon	Bon	Moyen	Mauvais	Moyen
Moments de Zernike	Bon	Bon	Bon	Bon	Mauvais	élevé
Moments de Chebyshev	Bon	Bon	Bon	Bon	Mauvais	élevé
Fourier 1-D	Bon	Bon	Bon	Mauvais	Mauvais	Moyen
Fourier région	Bon	Bon	Bon	Bon	Moyen	élevé
Signature système complexes	Mauvais	Mauvais	Mauvais	Moyen	Bon	faible
Signature distance centrale	Bon	Bon	Bon	Moyen	Bon	faible
Signature tangente	Bon	Bon	Bon	Mauvais	Bon	faible
Signature courbure	Bon	Bon	Bon	Mauvais	Bon	faible
Signature surface	Bon	Bon	Bon	Bon	Bon	faible
Signature surface triangle	Bon	Bon	Bon	Bon	Moyen	faible
Signature longueur de corde	Bon	Bon	Bon	Mauvais	Mauvais	faible

Tableau 3.2: L'invariance et la résistance des descripteurs de forme

- Elle doit être capable d'identifier les objets perceptuellement similaires qui ne sont pas mathématiquement identiques.
- Une mesure de similarité doit être robuste contre les différents types de distorsions.
- Elle doit être invariante aux différents types de transformations géométriques.

3.5.1 Distance métrique

En vision par ordinateur, la mesure de similarité est effectuée en calculant la distance entre les caractéristiques de chaque forme, en attribuant une faible valeur de distance aux objets similaires, une distance métrique est définie comme suit :

Un espace R^N est dit métrique si pour chaque deux éléments x et y , nous avons une

distance $d(x, y)$ qui satisfait les propriétés suivantes [BBP00] :

- $d(x, y) \geq 0$ (non négative).
- $d(x, y) = 0$ seulement si $x = y$ (identité).
- $d(x, y) = d(y, x)$ (symétrie).
- d est continue.
- Une grande différence entre les formes doit générer une grande dissimilarité.
- $d(x, y) \leq d(x, y) + d(y, z)$ (inégalité triangulaire).

Un descripteur de forme est généralement représenté comme un ensemble de vecteurs, la distance de similarité est la distance entre ces vecteurs, soit $A = (a_0, a_1, \dots, a_{N-1})$ et $B = (b_0, b_1, \dots, b_{N-1})$, les descripteurs des deux formes à comparer :

3.5.1.1 Distance Minkowski

La distance Minkowski est définie comme suit :

$$d_p(A, B) = \left(\sum_{i=0}^{N-1} |a_i - b_i|^p \right)^{1/p} \quad (3.34)$$

Dans le cas où $p = 1$, $d_1(A, B)$ on l'appelle la distance de Manhattan d_{Man} entre deux vecteurs A, B . $d_{Man} = d_{11} + d_{12}$ (Figure 3.9.(a)).

$$d_{Man}(A, B) = \sum_{i=0}^{N-1} |a_i - b_i| \quad (3.35)$$

Si $p = 2$ on obtient la distance Euclidienne d_E , c'est la distance ordinaire entre deux vecteurs (Figure 3.9.(b)).

$$d_E(A, B) = \sqrt{\sum_{i=0}^{N-1} (a_i - b_i)^2} \quad (3.36)$$

si $p \rightarrow \infty$ on obtient la distance Chebyshev ou L_∞ .

$$L_\infty(A, B) = \max_{0 \leq i \leq N-1} \{|a_i - b_i|\} \quad (3.37)$$

3.5.1.2 Distance cosinus

Cette distance permet de mesurer la similarité entre deux vecteurs (A, B) de N dimensions en déterminant le cosinus de l'angle entre eux (Figure 3.9.(c)).

$$d_{\cos}(A, B) = 1 - \cos\theta = 1 - \frac{A \cdot B^T}{\|A\| \cdot \|B\|} = 1 - \frac{\sum_{i=0}^{N-1} a_i b_i}{\sqrt{\sum_{i=0}^{N-1} a_i^2} \sqrt{\sum_{i=0}^{N-1} b_i^2}} \quad (3.38)$$

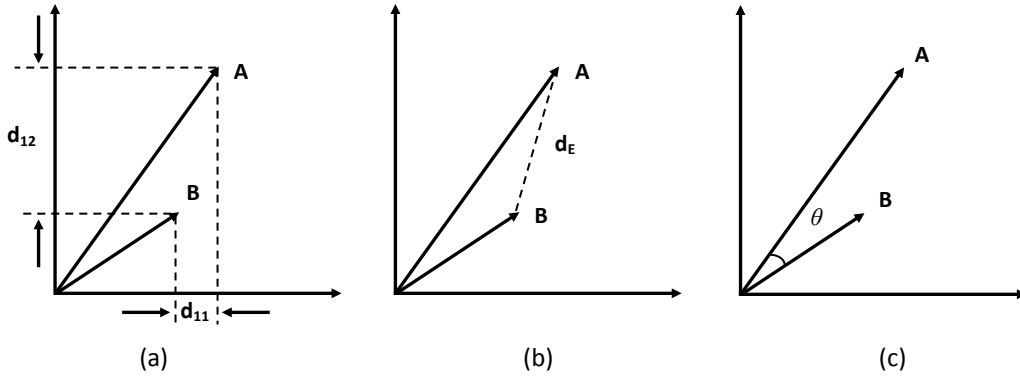


FIGURE 3.9: La représentation des distances Manhattan, Euclidienne et Cosinus

3.5.1.3 Distance statistique du Chi carré

La distance statistique du Chi carré χ^2 est définie comme suit :

$$d_{\chi^2}(A, B) = \frac{1}{2N} \sum_{i=0}^{N-1} \frac{(a_i - b_i)^2}{(a_i + b_i)} \quad (3.39)$$

3.5.1.4 Distance de Mahalanobis

La distance de Mahalanobis [Mah36] est basée sur la corrélation entre les variables afin de déterminer la similarité entre deux vecteurs. Contrairement à la distance euclidienne, elle prend en compte la covariance de la série de données. La distance de Mahalanobis donne un poids moins significatif aux composantes les plus dispersées. Elle est généralement utilisée pour détecter les données aberrantes et déterminer la cohérence de données.

$$d_M(A, B) = \sqrt{(A - B) \cdot C^{-1} \cdot (A - B)^T} \quad (3.40)$$

Où C est la matrice de covariance.

3.5.1.5 Distance de Hausdorff

La distance de Hausdorff [Ruc96] permet de mesurer l'éloignement de deux ensembles dans un espace métrique. La distance de Hausdorff entre deux vecteurs A et B est la plus longue distance entre un point arbitraire de A au point correspondant le plus proche de B (Figure 3.10). Cette distance n'est pas invariante aux transformations géométriques et sensible au bruit. L'avantage d'utiliser cette distance est que l'appariement partiel est possible.

$$d_H(A, B) = \max \left\{ \max_{b \in B} \left[\min_{a \in A} (d(a, b)) \right], \max_{a \in A} \left[\min_{b \in B} (d(a, b)) \right] \right\} \quad (3.41)$$

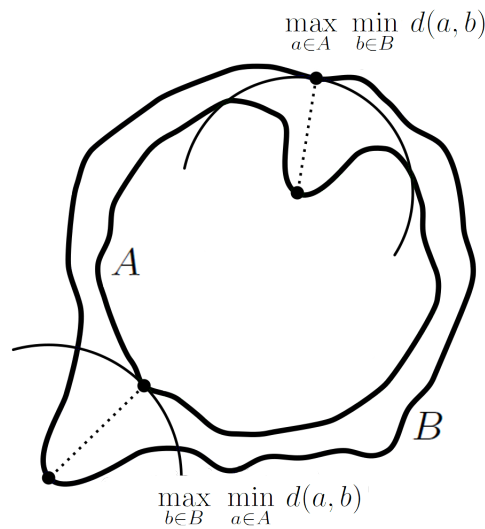


FIGURE 3.10: La distance de Hausdorff

3.5.2 Distance non métrique

Une distance est dite métrique si elle remplit les conditions suivantes (non négative, identité, symétrie, continue, inégalité triangulaire), dans la figure 3.11, l'homme et le centaure sont partiellement similaire, donc la distance entre l'homme et le centaure est petite, le même cas avec le centaure et le cheval, mais la distance entre le cheval et l'homme est grande : $d(\text{homme}, \text{centaure}) + d(\text{centaure}, \text{cheval}) < d(\text{homme}, \text{cheval})$, on voit que l'inégalité triangulaire n'est pas vérifiée. Cet exemple montre que la perception humaine de similarité est non métrique.

L'inégalité triangulaire n'est généralement pas vérifiée pour les distances de similarité qui sont robustes au bruit et les valeurs aberrantes, c'est pour cette raison que les distances non métriques sont plus puissantes et plus robustes par rapport aux distances métriques, elles sont utilisées principalement pour les bases de données multimédia et la recherche d'information.

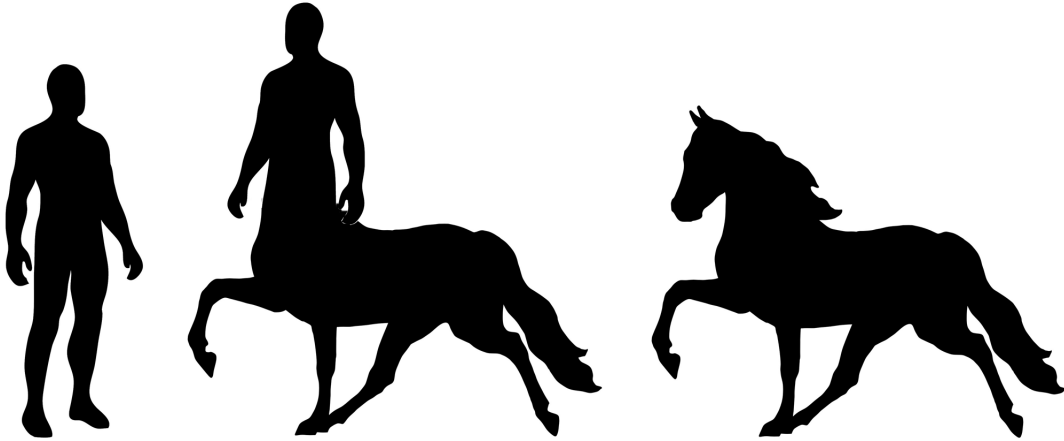


FIGURE 3.11: La perception humaine de similarité

3.5.2.1 L'appariement élastique non linéaire

L'appariement élastique non linéaire (Nonlinear elastic matching distance) est une distance non métrique qui ne respecte pas l'inégalité triangulaire. Nous avons deux formes qui sont similaires suivant la perception humaine, une de ces deux formes peut être une transformation rigide (translation, rotation, mise à l'échelle, etc.) de l'autre forme, dans ce cas, cette transformation peut être représentée mathématiquement où les distances métriques peuvent être utilisées, alors que dans les autres cas, cette transformation ne peut pas être représentée mathématiquement (transformation non-rigide, bruit, occultation, etc.) dans lequel les méthodes non métriques comme l'appariement élastique sont idéales.

On a $A = \{a_1, a_2, \dots, a_m\}$, $B = \{b_1, b_2, \dots, b_n\}$ deux ensembles de points ordonnés du contour, est f la correspondance entre tous les points de A et B de telle sorte qu'il n'existe aucun $a_1 < a_2$ où $f(a_1) > f(a_2)$. L'étendue $s(a_i, b_j)$ de $(a_i, f(a_i) = b_j)$ est égale à 1 si $f(a_{i-1}) = b_j$ ou $f(a_i) = b_{j-1}$, ou 0 autrement. L'appariement élastique non linéaire $NEM(A, B)$ est le minimum sur toutes les correspondances f de $\sum s(a_i, b_j) + d(a_i, b_j)$, où $d(a_i, b_j)$ représente la différence entre les tangentes à a_i et b_j [Vel01].

3.5.2.2 Distance k-médiane

La distance k-médiane entre deux vecteurs [JWG00](ou deux images représentées par des vecteurs) A et B est définie par :

$$d_k(A, B) = k - \text{medianes}\{|a_1 - b_1|, \dots, |a_N - b_N|\} \quad (3.42)$$

Où l'opérateur k-médiane donne comme résultat la k-ième valeur du vecteur de différence ordonnée.

3.6 Transformations

L'appariement de formes étudie la transformation entre les formes, afin de trouver la transformation qui minimise le coût de la mise en correspondance, et en conséquence le meilleur alignement entre les formes. Nous présentons dans cette section quelques transformations :

3.6.1 Translation

Une transformation à deux degrés de liberté [GS07], x et y les déplacements sur t_x et t_y respectivement.

$$t(x) = Rx + t \quad (3.43)$$

Ce qui s'écrit sous forme matricielle :

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ 1 \end{pmatrix} \quad (3.44)$$

3.6.2 Transformation rigide

La transformation rigide est une combinaison entre la translation et la rotation [Tal09], ce type de transformation conserve les distances et les angles. La transformation rigide peut être définie par la forme suivante :

$$t(x) = Rx + t \quad (3.45)$$

Ou par la forme matricielle :

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{pmatrix} + \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (3.46)$$

où t est un vecteur de translation, et R la matrice de rotation.

3.6.3 Similitude

En géométrie euclidienne, les similitudes sont des transformations à 4 degrés de liberté en 2D, c'est la combinaison entre les transformations rigides et la multiplication d'échelle par un rapport fixe [Tal09]. Les similitudes ne conservent pas les distances, mais elles

conservent les angles et les proportions de l'image.

$$t(x) = sRx + t, \quad s > 0 \quad (3.47)$$

3.6.4 Transformation affine

Une transformation géométrique entre les espaces affines, cette transformation conserve le parallélisme, elle est définie par :

$$t(x) = Ax + t \quad (3.48)$$

où A est une matrice qui représente une transformation linéaire [Mah06]

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (3.49)$$

3.6.5 Transformation projective

Les transformations projectives sont utilisées généralement pour représenter le passage à partir des objets 3D vers les images 2D en utilisant une caméra par exemple, en tenant compte aux effets de perspective lors de la composition de l'image. La transformation projective ajoute deux degrés de liberté à la transformation affine pour traiter les effets de la projection, le résultat est alors une transformation non-linéaire définie comme suit :

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (3.50)$$

3.7 Conclusion

Dans ce chapitre, nous avons présenté quelques techniques utilisées dans l'appariement de formes, nous avons commencé par la présentation des méthodes utilisées pour la représentation et la description de forme comme les caractéristiques géométriques simples et ensuite les descripteurs les plus complexes, et nous avons vu comment ces descripteurs peuvent être classés en trois types : les techniques basées contour, région et l'ensemble fini de points où nous avons cité pour chaque type quelques exemples des techniques les plus utilisées.

Ensuite, nous avons présenté les différentes distances et mesures de similarité entre les formes, et la différence entre les distances métriques et non métriques. Enfin, nous avons exposé quelques transformations géométriques qui peuvent être appliquées sur les formes.

Au chapitre suivant, nous allons introduire notre méthode d'appariement de formes basée sur la combinaison entre les algorithmes évolutionnaires inspirés du quantique et le descripteur shape context.

Approche évolutionnaire quantique pour l'appariement de formes

Nous présentons dans ce chapitre notre approche évolutionnaire quantique pour l'appariement de formes. Cette approche est basée sur l'appariement avec shape context et les algorithmes évolutionnaires quantiques. Nous exposons le fonctionnement détaillé de notre algorithme, la représentation quantique qui nous permet d'utiliser shape context, et la stratégie de recherche appliquée.

4.1	Introduction	70
4.2	Principes de l'algorithme évolutionnaire quantique pour l'appariement de formes	70
4.2.1	Descripteur de forme shape context	70
4.2.2	Représentation quantique	73
4.2.3	Fonction objectif	75
4.2.4	Stratégie de recherche	75
4.2.5	Algorithme QSC proposé	77
4.3	Conclusion	80

4.1 Introduction

L'appariement de formes est un problème central en vision par ordinateur, il est requis dans plusieurs domaines tels que l'imagerie médicale, la robotique, etc. Dans la littérature nous trouvons plusieurs techniques d'appariement de formes comme SC (Shape Context) [BMP00], IDSC (Inner-Distance Shape Contexts) [LJ07] et HF (Height Functions) [WBY⁺12], etc. Le descripteur shape context est une des techniques les plus populaires de l'appariement de formes grâce à sa puissance et sa robustesse. Des chercheurs ont essayé d'ajouter des contributions à ce descripteur afin de le rendre plus robuste à certains types de transformations et de déformations.

Dans ce travail, nous proposons une nouvelle approche d'appariement de formes basée sur la combinaison des algorithmes évolutionnaires quantiques et l'appariement par le descripteur shape context [MMF14]. Notre algorithme quantique d'appariement de formes (QSC) est fondé sur une représentation quantique pour coder les solutions possibles et une stratégie de recherche évolutionnaire quantique pour déterminer le meilleur appariement possible entre les formes.

Dans ce chapitre, nous commençons par une présentation de l'appariement de formes avec le descripteur shape context, les différentes étapes de sa construction, ses avantages, et ses points faibles. Ensuite, nous procédons à la présentation de notre approche proposée, la représentation quantique du problème et la stratégie de recherche utilisée. Enfin, nous présentons notre algorithme quantique avec une description de chaque étape dans l'algorithme.

4.2 Principes de l'algorithme évolutionnaire quantique pour l'appariement de formes

4.2.1 Descripteur de forme shape context

Shape context est un descripteur de forme très puissant introduit par Belongie et al. [BMP00], ce descripteur est parmi les méthodes basées sur un ensemble fini de points, dans l'idée principale est de trouver la meilleure correspondance entre les points extraits du contour des deux formes afin de minimiser le coût globale de l'appariement entre les deux formes.

La première étape consiste à la détection du contour de la forme, ensuite l'échantillonnage du contour et la sélection d'un ensemble de s points (Figure 4.1 (b)). Pour construire l'histogramme de la distribution spatiale des autres $(s - 1)$ points par rapport au point P_i , des vecteurs de P_i vers tous les autres points sont tracés (Figure 4.1 (c)) dans un

système de coordonnées polaires logarithmiques divisé uniformément par des cases pour rendre le descripteur plus sensible aux points plus proches [BMM06] (Figure 4.1 (f)). La longueur r et l'orientation θ de ces vecteurs sont quantifiées pour créer l'histogramme h_i qui représente le nombre de points appartenant à chaque case (Figure 4.1 (d)). L'ensemble des histogrammes de tous les s points de la forme représente le descripteur shape context (Figure 4.1 (e)).

$$h_i(k) = \#\{Q \neq P_i : (Q - P_i) \in \text{bin}(k)\} \quad (4.1)$$

Le descripteur shape context est ensuite utilisé pour trouver la correspondance entre les points des deux formes. Pour ce faire, nous calculons d'abord le coût de l'appariement $C_{i,j}$ entre chaque point i de la première forme, et le point j dans la deuxième forme en utilisant la statistique de Chi carré χ^2 .

$$C_{i,j} = \frac{1}{2} \sum_{k=1}^K \frac{[g(k) - h(k)]^2}{g(k) + h(k)} \quad (4.2)$$

Le résultat est une matrice qui contient le coût de l'appariement pour chaque paire de points dans les deux formes. L'objectif maintenant est la recherche de l'ensemble des paires qui minimise le coût total de l'appariement, alors le problème d'appariement devient un problème d'affectation dans un graphe biparti qui peut être résolu en temps polynomial par l'algorithme hongrois.

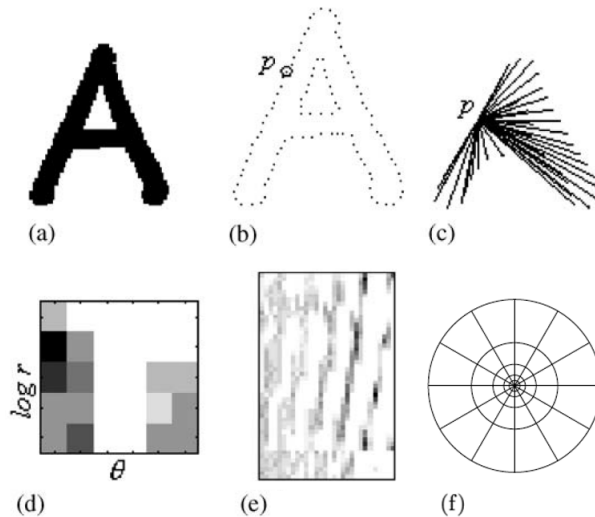


FIGURE 4.1: Le descripteur shape context

Après avoir obtenu les correspondances entre l'ensemble des points, le modèle thin-plate-splines (TPS)(Figure 4.2) [Boo89] est utilisé itérativement pour estimer une transformation entre les deux formes. Enfin, la distance de dissimilarité entre les deux formes est

donnée par l'équation suivante :

$$D = aD_{ac} + D_{sc} + bD_{be} \quad (4.3)$$

où D_{sc} est la distance de shape context, D_{be} l'énergie de flexion (bending energy), D_{ac} distance de l'apparence (appearance distance), a et b sont des poids ($a = 1.6$, $b = 0.3$ dans [BMM06]).

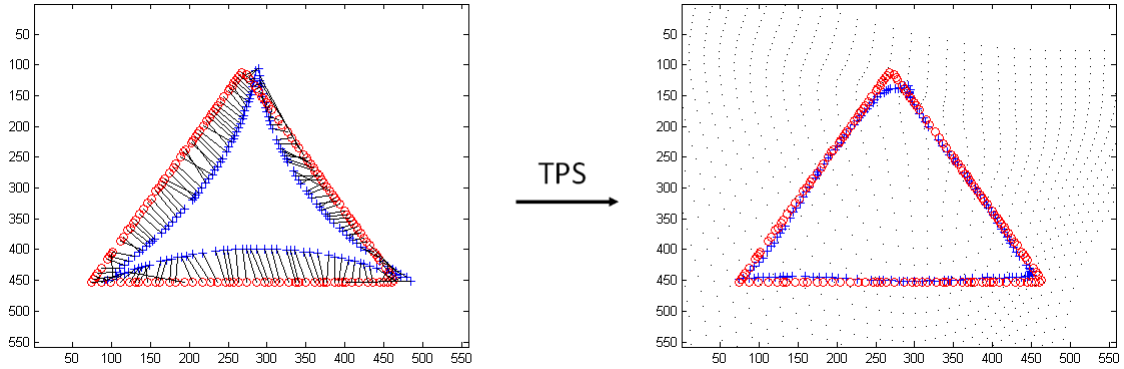


FIGURE 4.2: La transformation thin-plate-splines (TPS)

Un algorithme d'appariement de formes doit être robuste aux transformations géométriques comme la translation, la rotation et la mise à l'échelle et les déformations non-linéaires comme le bruit, l'articulation, l'occultation. L'invariance à la translation est obtenue directement dans shape context puisque la mesure de la distance entre les points est toujours relative au point central pour chaque histogramme $h(k)$. Pour atteindre l'invariance à la mise à l'échelle nous normalisons toutes les distances radiales par la distance moyenne entre les m^2 paires de points dans la forme [BMP02].

La faiblesse de shape context réside dans l'invariance à la rotation [BAKS12], celle-ci dépend de la tangente en chaque point du contour ; pour pouvoir calculer la tangente en un point, il faut que les points du contour sont organisés dans l'ordre (le séquençage dans le contour) [YW07]. Ling et al.[LJ07] ont également utilisé la programmation dynamique avec le séquençage des points du contour et une nouvelle mesure de l'angle relatif entre les points nommé (inner-angle) pour obtenir une meilleure invariance à la rotation.

Imposer un ordre sur les points représente une contradiction à l'esprit fondamental de shape context qui consiste à choisir les points aléatoirement et sans classement. Ce principe est un avantage supplémentaire de shape context qui lui permet d'être appliqué directement sur les points sans prétraitement comme l'organisation perceptuelle des points du contour [YW07].

L'organisation perceptuelle des points du contour empêche shape context d'être appliqué à d'autres cas que la représentation de forme basée contour, par exemple la technique

basée sur un squelette des points [Yan05] [XHS08].

4.2.2 Représentation quantique

L'appariement entre deux formes A et B est réalisé en trouvant la meilleure correspondance entre les points $A = \{a_1, a_2, \dots, a_s\}$ et $B = \{b_1, b_2, \dots, b_s\}$. Pour obtenir de bons résultats d'appariement avec la méthode de shape context, il faut utiliser un grand nombre de points pour chaque forme, ce qui conduit à un temps de calcul plus important. Avec notre approche nous avons besoin d'un nombre plus petit de points en essayant de trouver la configuration des points la plus adéquate à partir d'un ensemble plus grand de points, ce qui conduit aux meilleurs résultats d'appariement.

Chaque forme F est représentée par s points p_1, p_2, \dots, p_s .

$$F = \begin{pmatrix} x_1 & x_2 & \dots & x_s \\ y_1 & y_2 & \dots & y_s \end{pmatrix} \quad (4.4)$$

Une représentation binaire est utilisée pour indiquer les points qui vont être sélectionnés pour le calcul de shape context et les points qui vont être ignorés pour chaque itération. Chaque entrée dans le vecteur (voir figure 4.3) représente un point du contour, si la valeur de l'entrée est 1 alors le point est sélectionné, sinon le point est ignoré.

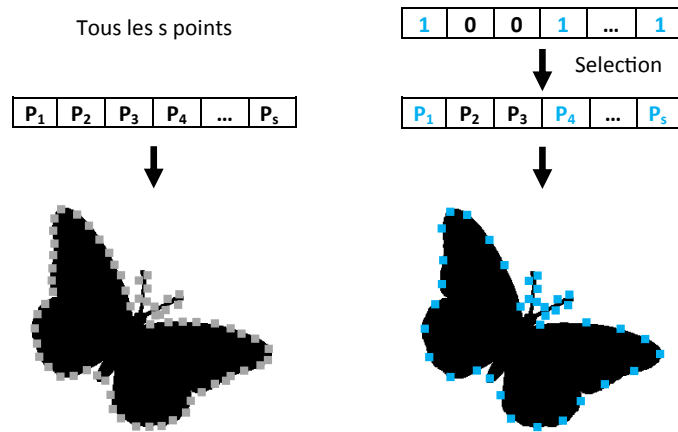


FIGURE 4.3: La sélection des points

Le descripteur shape context est très robuste avec de très bonnes propriétés d'invariance, il est fort contre la translation et la mise à l'échelle, cependant son invariance à la rotation est encore faible [YW07]. Pour résoudre ce problème, nous avons utilisé les AEQs pour estimer l'orientation qui optimise notre fonction objectif, ce qui conduit à de bons résultats d'appariement sur les formes avec des changements d'orientation.

Pour chaque chromosome, nous utilisons 5 bits pour contrôler l'orientation de la forme (Figure 4.4), le premier bit est utilisé pour le retournement vertical, si la valeur du bit est 1 alors nous allons appliquer un retournement mais si la valeur du bit est 0 alors nous n'appliquons pas de retournement. De la même manière, nous utilisons le deuxième bit pour le retournement horizontal. Pour la rotation, nous utilisons 3 bits afin de contrôler l'orientation sur $[0, 2\pi]$ divisé par 2^3 .

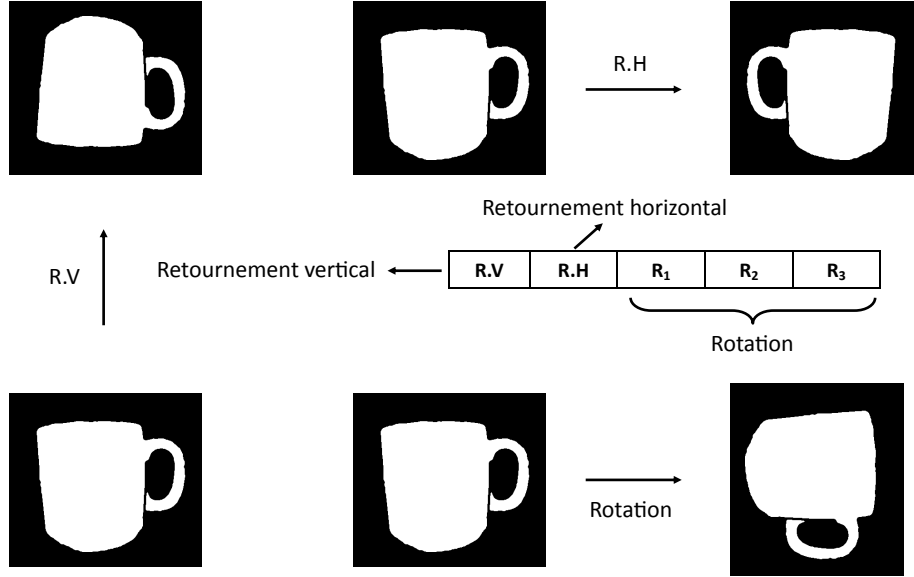


FIGURE 4.4: Contrôler l'orientation de la forme

La représentation quantique est réalisée à l'aide d'un registre quantique q de m qubits où $m = 2s + 5$, chaque une des deux formes est représentée par s qubits (chaque qubit représente l'état d'un point), et 5 qubits pour contrôler l'orientation de la forme.

$$q = \begin{pmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_m \\ \beta_1 & \beta_2 & \cdots & \beta_m \end{pmatrix} \quad (4.5)$$

L'avantage de cette représentation quantique est qu'elle ne représente pas seulement une solution, mais toutes les solutions possibles, grâce au phénomène de la superposition quantique, mais notons bien que la mesure donne seulement une seule solution.

Chaque registre quantique q représente un individu (chromosome) dans notre population quantique $Q(t)$ tel que $Q(t) = \{q_1^t, q_2^t, \dots, q_n^t\}$ où t représente la génération actuelle de la population et n le nombre d'individus.

$$Q(t) = \begin{pmatrix} q_1^t \\ q_2^t \\ \vdots \\ q_n^t \end{pmatrix} = \begin{pmatrix} \alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,m} \\ \beta_{1,1} & \beta_{1,2} & \cdots & \beta_{1,m} \\ \alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{2,m} \\ \beta_{2,1} & \beta_{2,2} & \cdots & \beta_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{n,1} & \alpha_{n,2} & \cdots & \alpha_{n,m} \\ \beta_{n,1} & \beta_{n,2} & \cdots & \beta_{n,m} \end{pmatrix} \quad (4.6)$$

Le résultat de l'opération de mesure de $Q(t)$ est une population binaire $P(t)$ tel que $P(t) = \{X_1^t, X_2^t, \dots, X_n^t\}$, où X_j^t est un vecteur binaire de m bits qui représente une solution (une configuration des points).

$$P(t) = \begin{pmatrix} X_1^t \\ X_2^t \\ \vdots \\ X_n^t \end{pmatrix} = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,m} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,m} \end{pmatrix} \quad (4.7)$$

4.2.3 Fonction objectif

Pour chaque deux formes A et B représentées par des points extraits du contour $A = \{a_1, a_2, \dots, a_s\}$ et $B = \{b_1, b_2, \dots, b_s\}$. La matrice de coût (cost matrix) représente le coût de l'appariement $C_{i,j}$ entre chaque point a_i de la première forme, et le point b_j dans la deuxième forme (Figure 4.5).

Le coût de l'appariement entre les deux formes est obtenu en trouvant les paires de points qui minimisent le coût total de l'appariement, ce qu'est réalisé en trouvant la permutation π qui minimise la fonction H en utilisant l'algorithme hongrois.

$$H(\pi) = \sum_{i=1}^s C(a_i, b_{\pi(i)}) \quad (4.8)$$

Chaque individu représente une version différente de shape context, donc une matrice de coût différente. Notre valeur de fitness est la valeur minimum de H dans tout l'espace de recherche.

4.2.4 Stratégie de recherche

Notre algorithme évolutionnaire quantique est un processus itératif dans l'objectif est de trouver la meilleure solution dans un espace de recherche, la convergence de l'algorithme

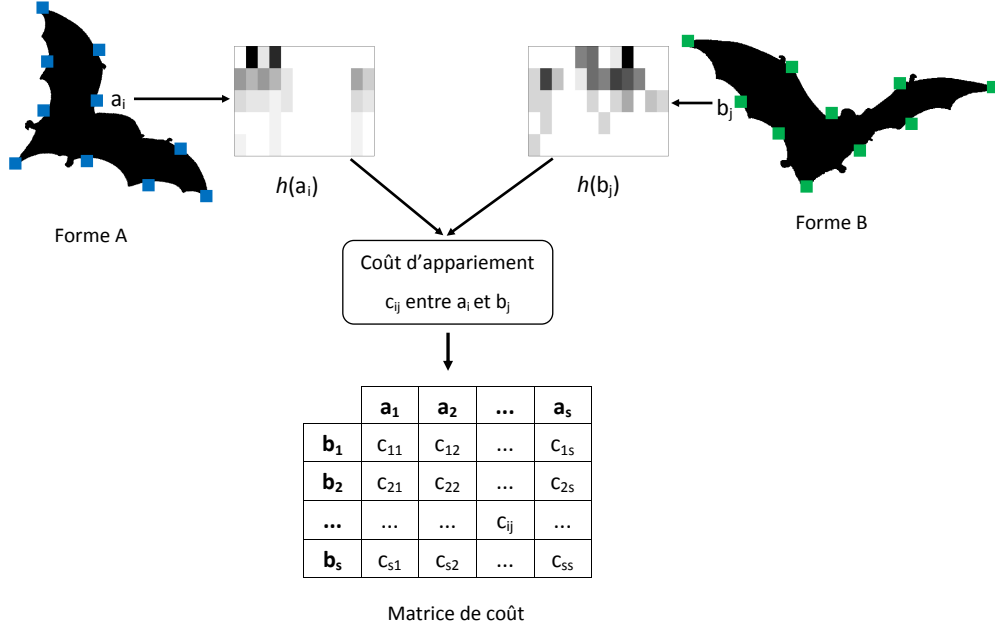


FIGURE 4.5: La matrice de coût

peut être atteinte en se basant sur deux principes : la diversification et le renforcement [Tal09]. La diversification cherche à explorer tout l'espace de recherche afin de trouver des nouvelles solutions, tandis que le renforcement permet de raffiner la recherche, en explorant les zones proches à la meilleure solution actuelle. Un bon équilibre entre les deux phénomènes doit être respecté ; le résultat d'un taux de diversification élevé est une recherche aléatoire, alors qu'avec un taux de renforcement élevé on ne peut pas explorer suffisamment l'espace de recherche (optimum local).

Selon Han et Kim [HK00] l'opérateur de rotation (interférence) quantique $U(\theta)$ est suffisant pour guider l'algorithme vers la convergence et que la diversité est assurée par la représentation probabiliste de qubit (Figure 4.6).

$$\begin{pmatrix} \alpha' \\ \beta' \end{pmatrix} = U(\theta) \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad (4.9)$$

Où α' et β' représentent le qubit produit de la rotation $U(\theta)$ avec le qubit de la génération précédente, θ représente l'angle de rotation, $U(\theta)$ peut être représentée sous la forme matricielle suivante :

$$U(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \quad (4.10)$$

Les opérateurs génétiques (croisement, mutation) peuvent être utilisés, mais ils ne sont pas nécessaires, et si les probabilités de mutation et de croisement sont élevées la performance de l'algorithme peut être diminuée notablement [HK00].

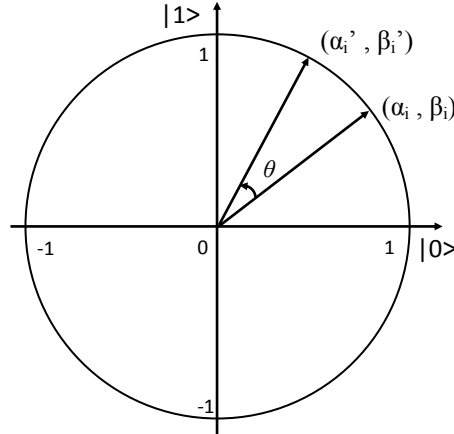


FIGURE 4.6: L'opérateur de rotation quantique

Dans notre algorithme, nous allons employer l'opérateur de rotation quantique comme le seul opérateur de mise à jour, nous nous basons sur la méthode décrite dans [HK02]. Le tableau 4.1 montre la valeur de l'angle de rotation θ obtenue par rapport à la valeur de bit x_i de l'individu X et la valeur de bit b_i de la meilleure solution b , où $f()$ représente la valeur de fitness de l'individu.

x_i	b_i	$f(X) \leq f(b)$	θ
0	0	faux	θ_1
0	0	vrai	θ_2
0	1	faux	θ_3
0	1	vrai	θ_4
1	0	faux	θ_5
1	0	vrai	θ_6
1	1	faux	θ_7
1	1	vrai	θ_8

Tableau 4.1: Le tableau de référence

4.2.5 Algorithme QSC proposé

Dans cette section, nous présentons un pseudo-code de notre algorithme quantique pour l'appariement de formes, ensuite nous décrivons de manière détaillée chaque étape de l'algorithme.

Données : Les deux formes (requête et cible)

Résultat : La distance d'appariement entre les deux formes

début

$t \leftarrow 0$;

(i) Extraire s points à partir du contour de chaque forme;

(ii) Générer la population initiale $Q(t)$ de n individus composé de m qubits;

tant que *Non convergence* **faire**

$t \leftarrow t+1$;

(iii) Produire $P(t)$ par la mesure de la population quantique $Q(t)$;

(iv) Réparer les individus de la population binaire $P(t)$;

(v) Sélectionnez la configuration des points pour chaque individu;

(vi) Construire descripteur shape context pour chaque individu;

(vii) Calculer le coût de l'appariement entre les deux formes;

(viii) Sauvegarder la meilleure solution de $P(t)$ dans $B(t)$;

(ix) Mettre à jour la population quantique $Q(t)$;

fin

fin

Algorithme 4 : L'algorithme QSC

- i). La première étape consiste à extraire s points à partir du contour des deux formes, ces points sont sélectionnés au hasard et ne représentent pas des points clés dans la forme, le descripteur shape context n'impose pas le tri des points, donc aucune information relative à l'ordre des points n'est nécessaire.
- ii). Dans la phase d'initialisation, nous générons la première population quantique $Q(0)$ qui est composée de n individus de m qubits, chaque qubit est initialisé avec la probabilité de ($\alpha_{i,j}^0 = 1/\sqrt{2}$ et $\beta_{i,j}^0 = 1/\sqrt{2}$) pour $i = 1 \cdots m$ et $j = 1 \cdots n$.

$$|\psi_{q_j^0}\rangle = \sum_{k=1}^{2^m} \frac{1}{\sqrt{2^m}} |X_k\rangle \quad (4.11)$$

- iii). L'opération de mesure prend la population quantique $Q(t)$ et donne comme résultat une population binaire $P(t)$, cette opération consiste à évaluer les probabilités de $|\alpha_i^t|^2$ et $|\beta_i^t|^2$ de chaque individu quantique q_j^t , $j = \{1, 2, \dots, n\}$ où chaque individu de la population binaire représente une configuration de points ou une solution, notons bien que nous utilisons la méthode de mesure probabiliste de Han et kim [HK00].
- iv). Pour construire le descripteur shape context, le nombre de points doit être identique

pour les deux formes, afin de créer ensuite une matrice de coût carré, donc il faut s'assurer que les solutions mesurées à partir des individus quantiques sont valides. Pour cette raison, un processus de réparation est employé sur chaque individu binaire X_j^t qui consiste à imposer un nombre fixe k où ($k \leq s$) de points à être sélectionnés, ce qui est réalisé en contrôlant le nombre de bits avec la valeur de 1 à chaque individu de la population $P(t)$.

En général, le résultat des opérations de mise à jour génétique quantique est un nombre différent de points sélectionnés pour chaque individu, donc en rendant leur nombre égal à k on peut garantir d'avoir toujours le même nombre de points sélectionnés pour les formes que nous sommes en train de mettre en correspondance, et que les résultats seront également plus cohérents et uniformes sur tout l'espace de recherche.

- v). Le processus de sélection consiste à choisir les points qui seront utilisés pour construire le descripteur shape context en se basant sur les individus binaires réparés. Chaque bit de l'individu binaire représente un point du contour, si la valeur de bit est 1, alors le point correspondant sera utilisé pour construire shape context, mais si la valeur de bit est 0 alors le point correspondant ne sera pas choisi pour cette itération.
- vi). Dans cette étape, nous passons à la construction de descripteur shape context. Chaque individu de la population binaire $P(t)$ représente une solution ou une configuration différente de points, chaque configuration de points est ensuite utilisée pour construire une version différente de shape context pour les deux formes, ensuite la matrice de coût (cost matrix) est calculée en utilisant le shape context des deux formes.
- vii). Notre algorithme essaie de trouver la configuration de points qui réduit le coût total d'appariement. Chaque version de shape context générée à l'étape précédente sera utilisée pour trouver le coût d'appariement minimum entre les deux formes. Ce coût d'appariement est considéré comme la valeur de fitness qui sera utilisée pour évaluer notre population quantique.
- viii). La meilleure solution obtenue par chaque individu est enregistrée dans b_j^t tel que $B(t) = \{b_1^t, b_2^t, \dots, b_n^t\}$. La meilleure solution dans $B(t)$ est sélectionnée, si cette dernière représente une amélioration par rapport à b , la valeur de b est remplacée par celle de $B(t)$. Notons bien que b est la meilleure solution globale (voire figure 4.7), elle sera utilisée ensuite pour guider les futures mises à jour de la population.

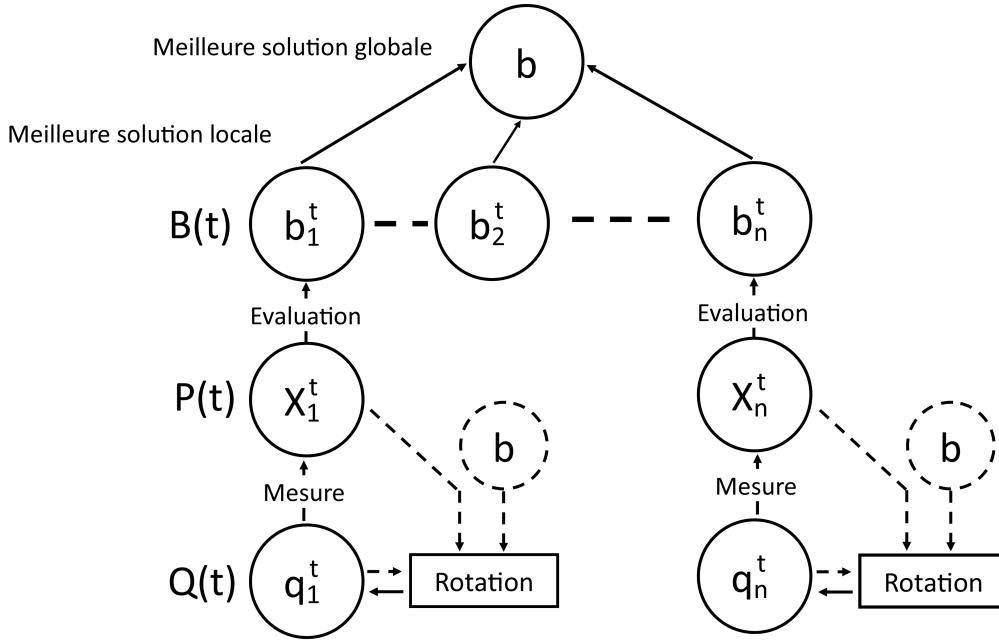


FIGURE 4.7: La sélection de la meilleure solution globale

ix). A la fin de chaque itération, nous mettons à jour notre population en utilisant l'opérateur de rotation quantique.

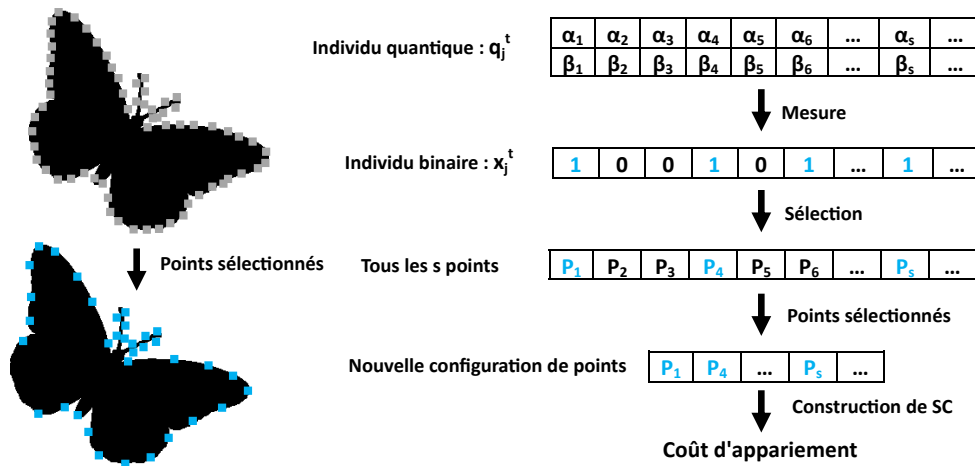


FIGURE 4.8: L'algorithme QSC

4.3 Conclusion

Nous avons présenté dans ce chapitre une nouvelle approche évolutionnaire quantique pour l'appariement de forme, notre approche est fondée principalement sur une combinaison entre les notions riches de l'informatique quantique ainsi que la puissance et la robustesse du descripteur shape context.

Nous avons proposé d'abord une représentation quantique probabiliste de l'espace de recherche par une population quantique qui permet de couvrir toutes les solutions éventuelles (théoriquement un seul individu quantique peut représenter toutes les solutions possibles). Nous avons proposé ensuite une fonction objectif (fitness) qui permet d'évaluer la population quantique, ainsi qu'une stratégie de recherche basée sur l'opérateur de rotation quantique pour manipuler cette population quantique afin d'atteindre les meilleures solutions en explorant tout l'espace de recherche.

Evaluation de l'approche proposée

Dans le chapitre précédent, nous avons présenté notre approche évolutionnaire quantique pour l'appariement de formes. Dans ce chapitre, nous allons présenter quelques résultats expérimentaux obtenus suite à l'évaluation des performances de notre approche proposée.

5.1	Introduction	83
5.2	Expérimentations et résultats	83
5.2.1	Matériels et logiciels utilisés	83
5.2.2	Protocole d'évaluation	84
5.2.3	Etude de l'effet de θ sur la convergence de l'algorithme	84
5.2.4	Etude de l'effet de nombre d'individus sur la convergence de l'algorithme	86
5.2.5	Recherche d'images	88
5.2.6	Temps de calcul	100
5.3	Conclusion	102

5.1 Introduction

Dans le chapitre précédent, nous avons exposé notre algorithme quantique pour l'appariement de formes, qui est basé sur la théorie quantique et le descripteur shape context. Théoriquement, cette combinaison possède de grands potentiels, cependant une évaluation rigoureuse doit être appliquée afin de connaître son apport vis-à-vis des autres méthodes.

Dans ce chapitre, nous présentons une étude expérimentale visant à évaluer la performance de l'approche QSC que nous avons proposée. Ces expérimentations sont faites en implémentant notre approche QSC et d'autres approches classiques à savoir SC et IDSC. Ces approches sont très connues et utilisées dans le but de comparer avec notre approche QSC.

Les bases d'images utilisées dans nos tests et évaluations sont très connues et très utilisées pour la validation des approches d'appariement de formes. Les résultats obtenus montrent nette supériorité de notre approche QSC par rapport aux approches classiques ce qui donne raison à notre choix.

5.2 Expérimentations et résultats

Nous présentons dans cette section, quelques résultats expérimentaux obtenus par notre méthode QSC et d'autres méthodes connues d'appariement de formes.

5.2.1 Matériels et logiciels utilisés

Notre algorithme a été implémenté en utilisant Matlab R2013a sur la plate-forme de Windows 8.1 64-bit avec un ordinateur de 6 Go de RAM et un processeur Intel Core i7 de 2.2 Ghz.

Nous avons opté pour l'utilisation de Matlab pour les raisons suivantes :

- Une librairie très riche des algorithmes de vision par ordinateur.
- Une très bonne documentation.
- Programmation plus rapide.
- La capacité de lire une large variété des formats d'images.
- La capacité d'appeler des bibliothèques externes.
- Une grande communauté d'utilisateurs.

5.2.2 Protocole d'évaluation

La recherche d'images est une des applications les plus importantes de l'appariement de formes, afin d'évaluer la performance de notre algorithme une mesure de performance efficace est nécessaire. Les mesures de performance sont généralement basées sur des techniques statistiques, la combinaison du rappel (recall) et la précision (precision) est peut-être la mesure de performance la plus utilisée dans la littérature [DG06].

Le rappel est le rapport entre le nombre des éléments pertinents trouvés et le nombre de tous les éléments pertinents de la base d'images, le rappel mesure la capacité de trouver des éléments pertinents dans une base d'images.

$$Recall = \frac{R_a}{R}, \quad \in [0, 1] \quad (5.1)$$

La précision est le rapport entre le nombre des éléments pertinents trouvés et le nombre d'éléments trouvés.

$$Recall = \frac{R_a}{A}, \quad \in [0, 1] \quad (5.2)$$

Où R est le nombre des éléments pertinents de la base d'images, A l'ensemble des réponses et R_a le nombre des éléments pertinents dans l'ensemble des réponses.

Lorsque la précision diminue le rappel augmente, en effet dans le processus d'essayer de récupérer tous les éléments pertinents à une requête, certains éléments non pertinents sont également récupérés. Le rappel et la précision sont utilisés ensemble pour indiquer l'efficacité d'un algorithme de recherche d'images.

5.2.3 Etude de l'effet de θ sur la convergence de l'algorithme

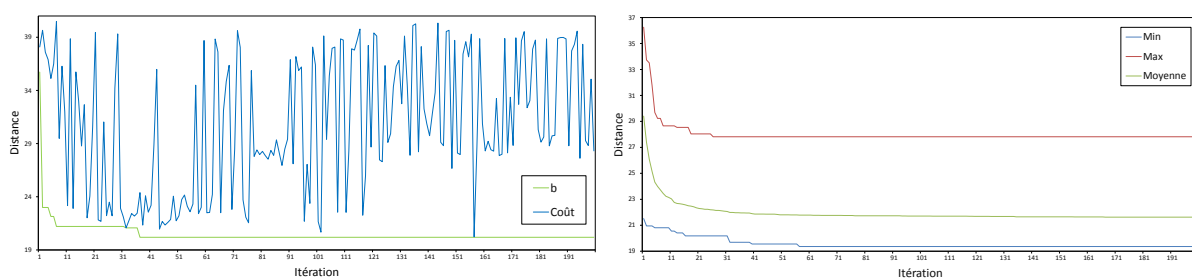
Dans ce test, nous essayons d'étudier l'effet du changement de la valeur de θ à la convergence de l'algorithme. Pour ce test, nous utilisons la configuration suivante : une population composée de 3 individus quantiques, 200 itérations, 80 points pour construire le descripteur shape context et la transformation TPS n'a pas été utilisée dans ce test. Les valeurs moyennes, min et max sont calculées à partir de 30 exécutions de notre algorithme.

Le test est appliqué sur deux images bat-1 et bat-2 (Figure 5.1) de la base d'image MPEG-7 CE-shape-1 [LLE00] qui se compose de 1400 images binaires de 70 catégories, 20 images par catégorie.



FIGURE 5.1: Image bat-1 à gauche et bat-2 à droite

a) Expérience avec $\theta = 0.04 \pi$:

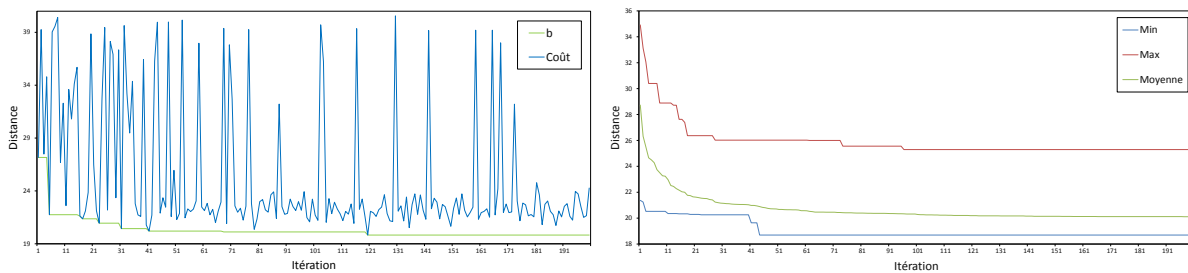


(a) L'évolution de fitness

(b) Les valeurs max, min et moyenne de fitness

FIGURE 5.2: Les résultats d'exécution avec $\theta = 0.04\pi$

b) Expérience avec $\theta = 0.02 \pi$:

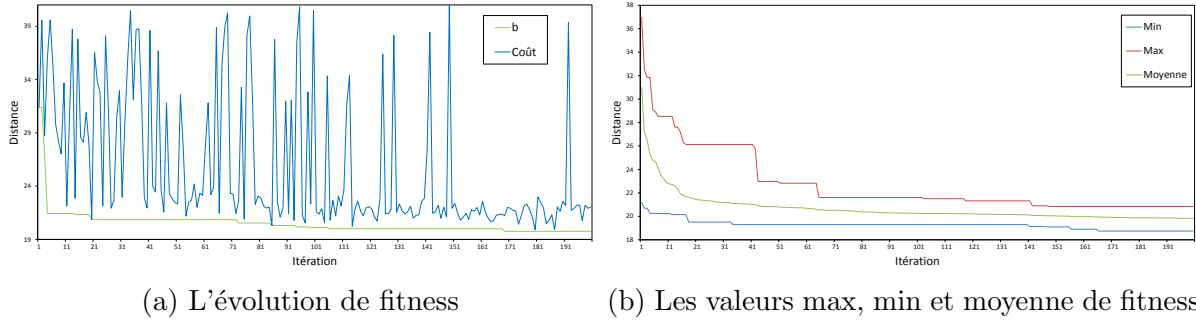


(a) L'évolution de fitness

(b) Les valeurs max, min et moyenne de fitness

FIGURE 5.3: Les résultats d'exécution avec $\theta = 0.02\pi$

c) Expérience avec $\theta = 0.01 \pi$:


 FIGURE 5.4: Les résultats d'exécution avec $\theta = 0.01\pi$

Thêta	0.04π	0.02π	0.01π
Meilleur résultat	19.36	18.69	18.74
Meilleur résultat (Moyenne)	21.63	20.11	19.82
Convergence (Moyenne)	55	95	164
Écart type	2.56	1.35	0.53

Tableau 5.1: Les résultats de l'étude de l'effet de thêta

On remarque, avec l'utilisation de la valeur 0.01π comme angle de rotation, que nous avons obtenu la meilleure performance de l'algorithme (meilleure valeur moyenne); la meilleure valeur d'écart type est aussi obtenue avec 0.01π , ce qui signifie que les résultats sont plus homogènes. Avec la valeur d'angle de rotation 0.04π , la convergence est plus rapide et dans certains cas nous avons obtenu de très bons résultats, mais dans plusieurs cas nous avons rencontré des situations d'optimum local, ce qui mène à des résultats hétérogènes.

5.2.4 Etude de l'effet de nombre d'individus sur la convergence de l'algorithme

Maintenant, nous essayons d'étudier l'effet du changement de nombre d'individus à la convergence de l'algorithme. Pour ce test, nous utilisons la configuration suivante : une valeur de thêta de 0.01π , 200 itérations, 80 points pour construire le descripteur shape context et la transformation TPS n'a pas été utilisée dans ce test. Les valeurs moyennes, min et max sont calculées à partir de 30 exécutions de notre algorithme. Le test est aussi appliqué sur les deux images bat-1 et bat-2 de la base d'image MPEG-7 CE-shape-1.

a) Expérience avec 2 individus :

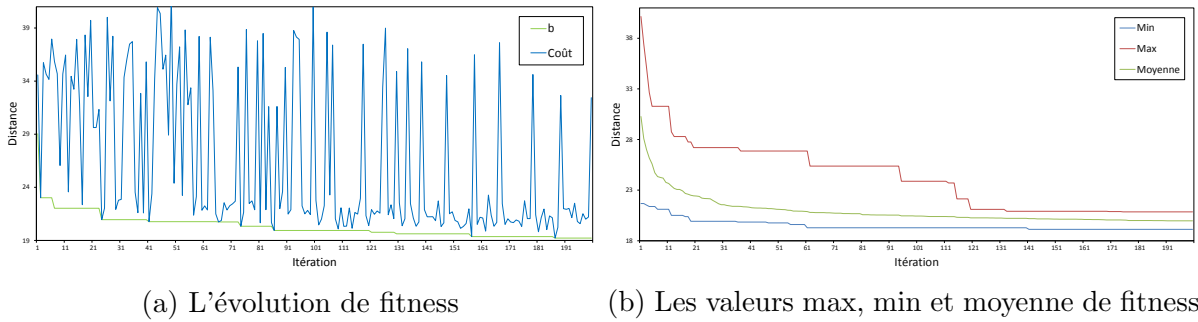


FIGURE 5.5: Les résultats d'exécution avec 2 individus

b) Expérience avec 4 individus :

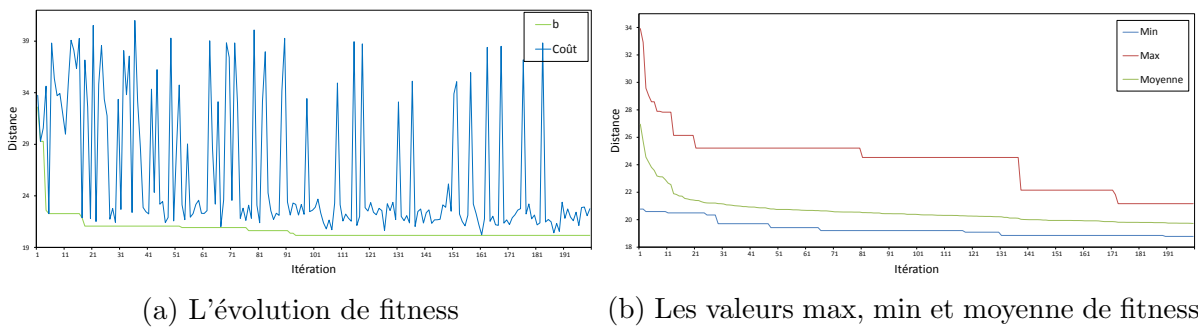


FIGURE 5.6: Les résultats d'exécution avec 4 individus

c) Expérience avec 8 individus :

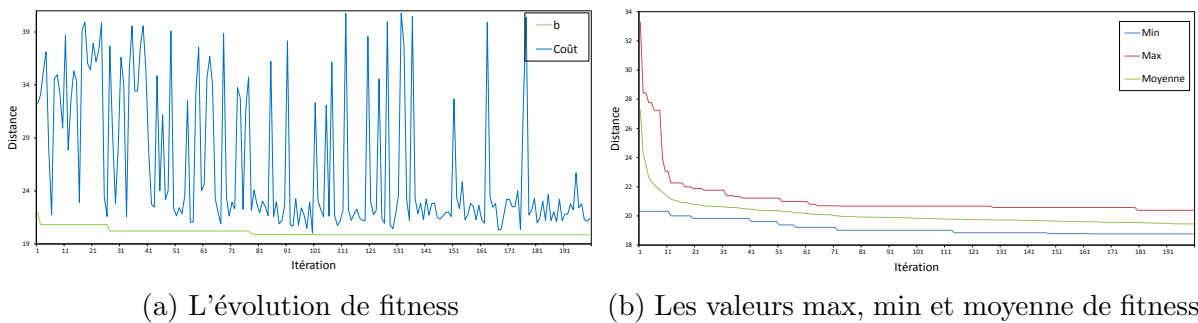


FIGURE 5.7: Les résultats d'exécution avec 8 individus

Nombre d'individus	2	4	8
Meilleur résultat	19.13	18.78	18.77
Meilleur résultat (Moyenne)	19.97	19.73	19.45
Convergence (Moyenne)	142	151	121
Écart type	0.41	0.54	0.42

Tableau 5.2: Les résultats de l'étude de l'effet de nombre d'individus

Nous remarquons l'augmentation des performances de l'algorithme avec l'augmentation de nombre des individus dans la population, cependant, cette amélioration reste insignifiante où nous avons pu obtenir de bons résultats avec 2 et 4 individus, ce qui confirme le principe de la superposition dans la théorie quantique qu'un seul individu est capable de représenter toutes les solutions possibles dans l'espace de recherche.

5.2.5 Recherche d'images

Dans cette section, nous présentons une évaluation de la performance de notre algorithme QSC dans la récupération de forme par rapport à d'autres techniques connues de récupération de forme telles que SC, IDSC et CSS. Les tests sont réalisés sur plusieurs bases d'images populaires comme MPEG-7, Kimia 25, Kimia 99 et COIL-100.

5.2.5.1 Expériences sur la base d'images MPEG-7

Ces tests sont appliqués sur la base d'images MPEG-7 CE-shape-1 [LLE00] qui contient 1400 images de 70 catégories, représentant plusieurs types de transformation comme la rotation, la mise à l'échelle, la translation et le bruit. Ce test prend une forme requête et calcule le pourcentage des formes trouvées qui appartiennent à la même classe que la requête (compris la forme requête) dans les 40 premières formes les plus similaires, sur le nombre total de formes par classe.

La configuration suivante est utilisé pour ce test : une population composée de 3 individus quantiques, 50 itérations, 40 points pour construire le descripteur shape context (pour SC, IDSC et QSC) et la transformation TPS n'a pas été utilisée dans ce test.

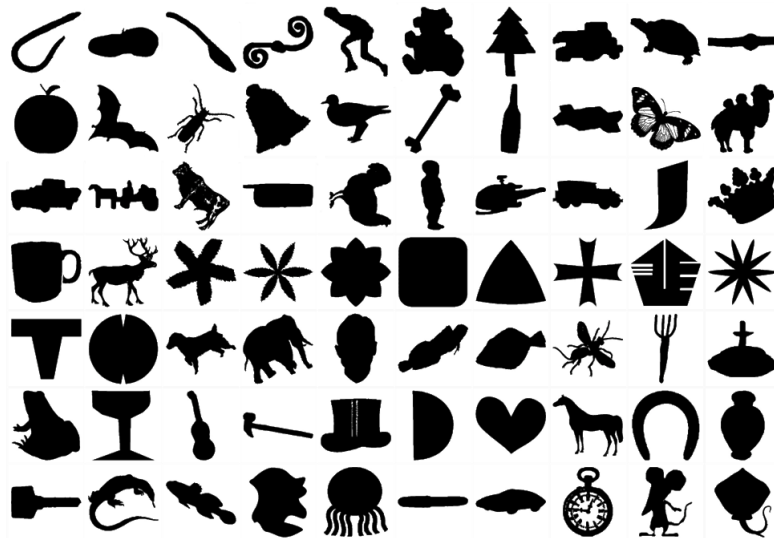







FIGURE 5.8: La base d'images MPEG-7 CE-shape-1

Requête	ALG	Top 10										Top 40
 bat-1	CSS											08
	SC											08
	IDSC											08
	QSC											16
 bird-2	CSS											11
	SC											10
	IDSC											06
	QSC											13
 fork-1	CSS											08
	SC											13
	IDSC											16
	QSC											19
 hammer-04	CSS											12
	SC											09
	IDSC											18
	QSC											17
 device0-19	CSS											20
	SC											10
	IDSC											15
	QSC											20






Requête	ALG	Top 10										Top 40
 elephant-1	CSS											09
	SC											08
	IDSC											08
	QSC											12
 chicken-1	CSS										13	
	SC										06	
	IDSC										15	
	QSC										12	
 apple-1	CSS										8	
	SC										20	
	IDSC										20	
	QSC										20	
 spring-1	CSS										05	
	SC										07	
	IDSC										08	
	QSC										16	
 frog-1	CSS										09	
	SC										09	
	IDSC										08	
	QSC										10	

Tableau 5.3: Les résultats de recherche de formes sur la base d'images MPEG-7

Méthode	Score (%)
Curvature scale space	51.5
Shape context	50
Inner-distance shape context	61
Quantum shape context	77.5

Tableau 5.4: Les résultats de recherche de formes sur la base MPEG-7

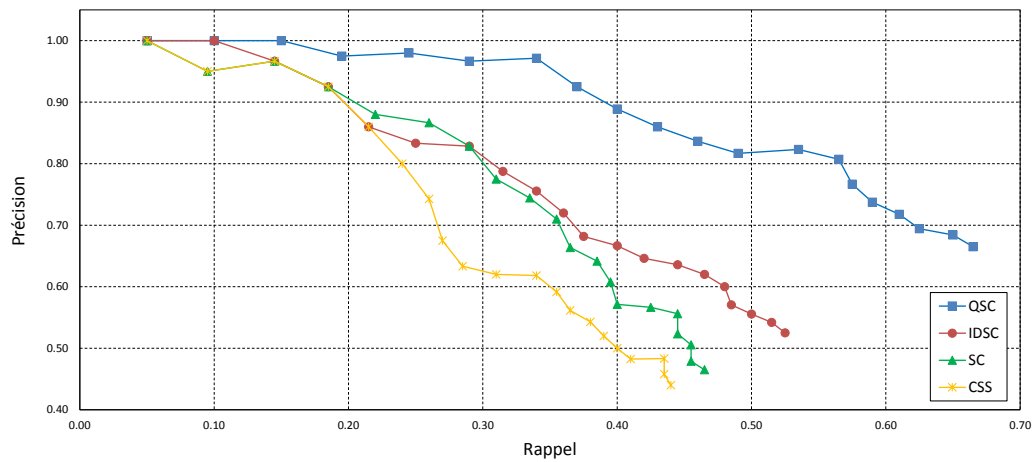


FIGURE 5.9: Le graphe de rappel-précision sur la base MPEG-7

Les résultats résumés dans le tableau 5.4 et le diagramme de rappel-précision (Figure 5.9) montrent l'efficacité de notre méthode proposée par rapport aux méthodes : SC, IDSC et CSS. Notre algorithme a réussi d'obtenir un bon taux de récupération sur les différents types de transformations, comme la rotation (bat-1 et hammer-4), le retournement (bird-2), le bruit (device0-19) et la mise à l'échelle (chicken-01).

5.2.5.2 Expériences sur la base d'images Kimia 25

Ces tests sont appliqués sur la base d'images Kimia 25 [SCTK98] qui contient 25 images de 6 catégories.

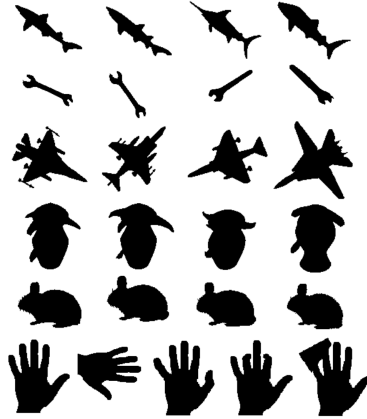


FIGURE 5.10: La base d'images Kimia 25

Ce test consiste à compter le nombre des premières, deuxièmes et troisièmes plus proches correspondances qui entrent dans la catégorie appropriée pour chaque requête. Cette expérience a été testée par shape context [BMP02], Sharvit [SCTK98], Gdalyahu [GW99], Inner-distance shape context [LJ07] et Chord context [MKJ08]. Pour ce test, nous avons utilisé la configuration suivante : une population composée de 3 individus quantiques, 10 itérations, 100 points pour construire le descripteur shape context (pour SC, IDSC et QSC) et la transformation TPS n'a pas été utilisée dans ce test. Les résultats obtenus sont reportés dans le tableau 5.5.

Méthode	Top 1	Top 2	Top 3
Sharvit	23/25	21/25	20/25
Gdalyahu	25/25	21/25	19/25
Shape context	25/25	24/25	22/25
Inner-distance shape context	25/25	24/25	25/25
Chord context	25/25	25/25	23/25
Quantum shape context	25/25	25/25	24/25

Tableau 5.5: Les résultats de recherche de formes sur la base d'images Kimia 25

Les résultats obtenus dans ce test, montrent clairement que notre algorithme proposé est plus performant par rapport aux autres algorithmes.

5.2.5.3 Expériences sur la base d'images Kimia 99

Dans ces tests, nous avons utilisé la base d'images Kimia 99 [SKK04] qui contient 99 images de 9 catégories, cette base consiste de plusieurs types de transformation comme la

rotation, la déformation et l'occultation partielle.

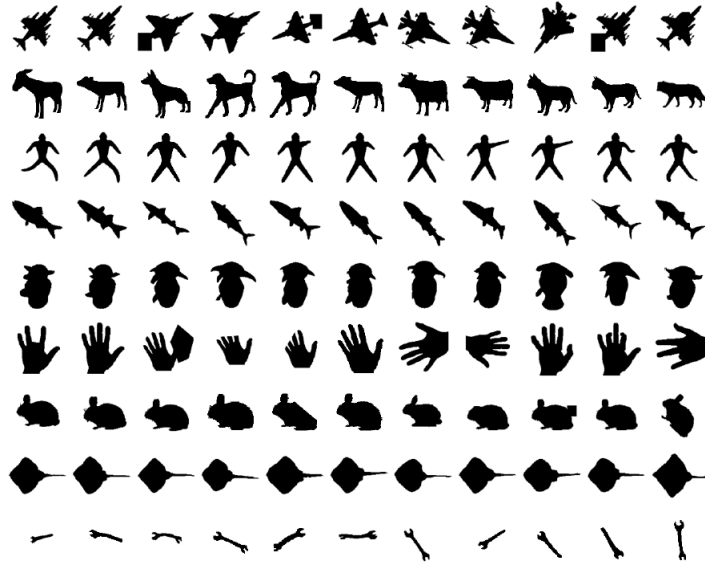

















































































































































FIGURE 5.11: La base d'images Kimia 99

Ce test consiste de compter le nombre des correspondances qui entrent dans la catégorie appropriée pour chaque requête dans les 15 plus proches formes. Nous allons appliquer ce test sur shape context [BMP02], inner-distance shape context [LJ07] et sur notre algorithme proposé. Le score de ce test est le pourcentage des formes trouvées qui appartiennent à la même classe que la requête (compris la forme requête) dans les 15 premières formes les plus similaires, sur le nombre total de formes par classe.

Pour ce test, nous avons utilisé la configuration suivante : une population composée de 3 individus quantiques, 20 itérations, 80 points pour construire le descripteur shape context (pour SC, IDSC et QSC) et la transformation TPS n'a pas été utilisée dans ce test. Les résultats obtenus sont reportés dans le tableau 5.7.

Requête	ALG	Top 10										Top 15
	SC											03
	IDSC											09
	QSC											11
	SC										02	
	IDSC										06	
	QSC										07	
	SC										08	
	IDSC										10	
	QSC										10	
	SC										01	
	IDSC										05	
	QSC										11	
	SC										11	
	IDSC										10	
	QSC										11	








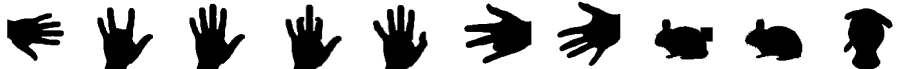

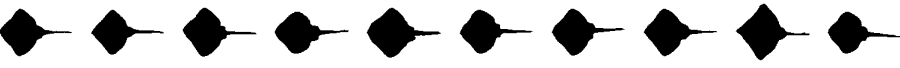
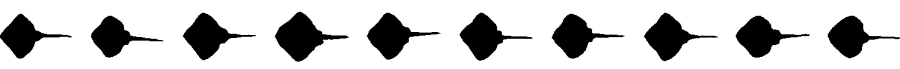









Requête	ALG	Top 10	Top 15
 quadruped	SC		11
	IDSC		11
	QSC		09
 hand	SC		01
	IDSC		10
	QSC		10
 fish2	SC		11
	IDSC		10
	QSC		11
 greebles	SC		11
	IDSC		09
	QSC		11
 fish	SC		11
	IDSC		07
	QSC		11

Tableau 5.6: Les résultats de recherche de formes sur la base d'images Kimia 99

Méthode	Score (%)
Shape context	63.64
Inner-Distance shape context	79.09
Quantum shape context	92.73

Tableau 5.7: Les résultats de recherche de formes sur la base Kimia 99

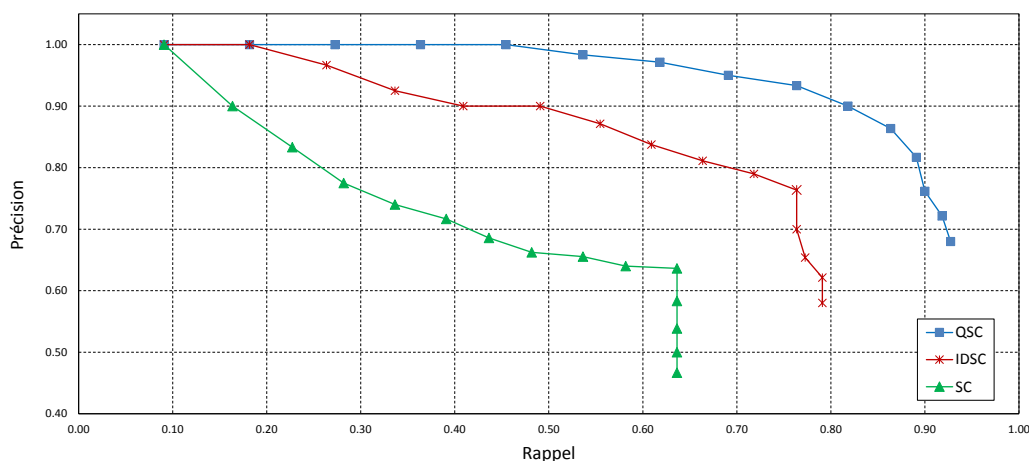







FIGURE 5.12: Le graphe rappel-précision sur la base Kimia 99

Les résultats obtenus dans le tableau 5.7 et le diagramme de rappel-précision (Figure 5.12) montrent la performance de notre méthode où nous avons obtenu un très bon taux de récupération sur plusieurs types de requêtes comme (hand et dude) qui représentent des formes articulées, la rotation (tool et plane) et l'occultation (plane et rabbit).

5.2.5.4 Expériences sur la base d'images COIL-100

La base COIL-100 (Columbia Object Image Library) [NNM96] contient 7200 images en couleurs prises à partir de 100 objets 3D avec des positions différentes. Les objets utilisés dans COIL-100 sont montrés dans la figure 5.13.

Requête	ALG	Top 10										Top 20
 obj8_60	SC											03
	IDSC											03
	QSC											05
 obj36_60	SC											06
	IDSC											04
	QSC											06
 obj74_60	SC											07
	IDSC											08
	QSC											08
 obj100_15	SC											02
	IDSC											03
	QSC											04
 obj2_60	SC											10
	IDSC											10
	QSC											10




























































































































































Requête	ALG	Top 10										Top 20
 obj99_60	SC											07
	IDSC											03
	QSC											07
 obj68_45	SC											04
	IDSC											04
	QSC											05
 obj19_0	SC											03
	IDSC											04
	QSC											04
 obj97_0	SC											04
	IDSC											04
	QSC											06
 obj73_60	SC											10
	IDSC											05
	QSC											10

Tableau 5.8: Les résultats de recherche de formes sur la base d'images COIL-100

Méthode	Score (%)
Shape context	56
Inner-distance shape context	48
Quantum shape context	65

Tableau 5.9: Les résultats de recherche de formes sur la base COIL-100

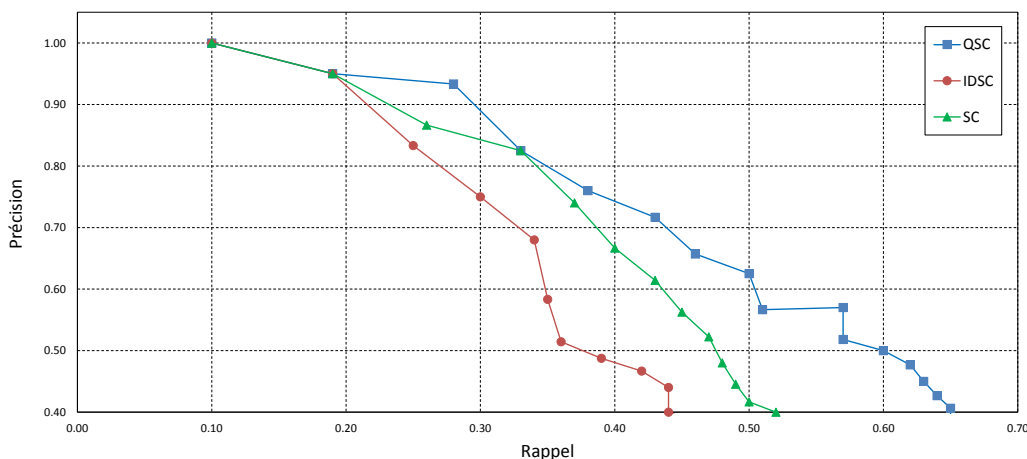


FIGURE 5.15: Le graphe rappel-précision sur la base COIL-100

Les résultats montrés dans le tableau 5.9 et le diagramme de rappel-précision (Figure 5.15) sont très encourageants en raison de la difficulté de ce test et du fait que nous avons utilisé seulement l'information de la forme de l'image. Nous avons obtenu le meilleur taux de récupération sur la plus part des requêtes testées, ce qui prouve l'efficacité de notre méthode proposée.

5.2.6 Temps de calcul

Dans cette section, nous présentons une comparaison de temps de calcul entre SC et notre algorithme QSC. Le premier test (Figure 5.16) consiste à mesurer le temps de calcul de l'appariement entre deux formes bat-1 et bat-2 de la base d'images MPEG-7. Le deuxième test (Tableau 5.10) consiste à mesurer le temps de calcul de la recherche d'une image dans toute la base d'images Kimia 99 qui contient 99 images. Le temps de calcul pour chaque test représente la valeur moyenne de 10 exécutions. La configuration utilisée pour chaque test est comme suit :

- SC : 40 points pour construire le descripteur shape context et 3 itérations de la transformation TPS [BMM06].
- QSC-1 : 40 points pour construire le descripteur shape context, une population

composée de 1 individu quantique, 10 itérations, la transformation TPS n'a pas été utilisée.

- QSC-2 : 40 points pour construire le descripteur shape context, une population composée de 2 individus quantiques, 10 itérations, la transformation TPS n'a pas été utilisée.
- QSC-4 : 40 points pour construire le descripteur shape context, une population composée de 4 individus quantiques, 10 itérations, la transformation TPS n'a pas été utilisée.

Les résultats dans la figure 5.16 et le tableau 5.10 montrent que le temps de calcul obtenu par SC est meilleur que le temps obtenu par notre algorithme QSC et que cette différence devient plus claire avec l'augmentation de nombre d'individus. Il est tout à fait clair que l'approche QSC proposée qui est réalisée d'une manière séquentielle prend plus de temps que l'approche SC, cette lourdeur est due à la multiplication des calculs de l'algorithme évolutionnaire quantique. Le temps de calcul de notre approche QSC peut être amélioré par l'utilisation d'une version parallèle à savoir Island Model [MAO06], ce qui est planifié dans nos futurs travaux.

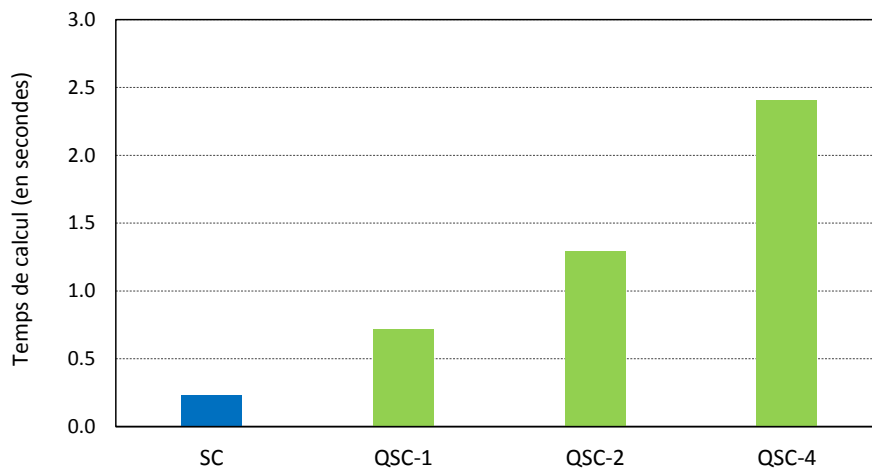


FIGURE 5.16: Une comparaison des temps de calculs pour SC et QSC

Test	Temps de calcul (en secondes)
SC	7.344
QSC-1	30.818
QSC-2	59.863
QSC-4	117.845

Tableau 5.10: les temps de calculs pour la recherche d'images sur la base Kimia 99

5.3 Conclusion

Dans ce chapitre, les résultats de l'évaluation expérimentale de notre approche ont été présentés, la performance de notre méthode a été comparée avec les méthodes SC, IDSC et CSS. Nous avons testé notre approche sur plusieurs bases d'images connues, comme la base MPEG-7 CE-shape-1, KIMIA 99 et COIL-100. Les résultats obtenus sont très prometteurs et montrent la faisabilité de notre approche proposée et sa capacité de trouver la meilleure correspondance entre les formes. Notre noyau quantique peut être également étendu par d'autres descripteurs et mesures de similarité qui présentent des possibilités d'améliorations ainsi que la capacité d'aborder d'autres types de problèmes.

Conclusion générale

Au cours de ce travail de magister, nous avons proposé une nouvelle approche d'appariement de formes basée sur les algorithmes évolutionnaires inspirés du quantique.

La première étape de notre travail a consisté à l'étude des différents descripteurs de formes et les diverses techniques d'appariement de formes existantes. Pour notre approche, nous avons proposé d'utiliser le descripteur shape context vue sa capacité discriminante et son invariance aux différentes transformations géométriques et le temps de calcul raisonnable. Ces avantages ont rendu shape context un des plus populaires descripteurs de formes.

Nous avons privilégié d'approcher le problème de l'appariement de formes en tant qu'un problème d'optimisation par le biais des méthodes métaheuristiques, dans ce sens nous avons opté pour les algorithmes évolutionnaires quantiques qui ont montré une amélioration importante par rapport aux algorithmes évolutionnaires classiques, grâce aux concepts inspirés de la mécanique quantique comme la superposition, l'interférence et l'observation probabiliste.

Notre algorithme quantique d'appariement de formes est basé sur une représentation quantique de l'espace de recherche. La nature probabiliste de la théorie quantique permet d'avoir une stratégie de recherche efficace, notre méthode emploie l'opérateur d'interférence quantique (rotation gate) en tant qu'un opérateur de mise à jour, ce qui permet de diriger la population vers les meilleures solutions. Le comportement de cet opérateur est contrôlé via l'angle de rotation afin de choisir la vitesse et la direction de recherche la plus adéquate.

Les résultats expérimentaux montrent l'intérêt de notre approche proposée, les tests ont été réalisés sur des bases d'images connues comme : la base MPEG-7, KIMIA et COIL-100, où nous avons testé notre approche contre les transformations géométriques (translation, mise à l'échelle, rotation), le bruit, l'articulation et l'occultation.

L'avantage de notre approche évolutionnaire quantique est qu'elle peut être étendue par d'autres types de descripteurs pour faire face à d'autres problèmes, ce qui rend notre approche idéale pour l'application à des problèmes spécifiques.

Certains points restent à améliorer, comme le choix des paramètres optimaux pour le fonctionnement de l'algorithme, tel que l'angle de rotation, le nombre des individus, le nombre d'itérations, ce choix est critique pour obtenir les meilleurs résultats. La possibilité

de considérer d'autres opérateurs pour la mise à jour de la population quantique. Réduire le temps de calcul élevé dans l'utilisation de notre algorithme quantique sur les grandes bases d'images. Trouver des solutions permettant d'éviter l'apparition d'optimum local dans certaines situations.

A partir du travail réalisé dans le cadre de ce mémoire, quelques perspectives peuvent être dégagées :

- Nous envisageons la réalisation d'une version distribuée de notre approche pour le but d'améliorer le temps d'exécution de l'algorithme.
- L'extension de l'algorithme avec d'autres descripteurs et des mesures de similarité pour améliorer la performance de notre approche dans les tâches de recherche de formes.
- Notre noyau quantique peut être aussi utilisé pour résoudre d'autres problèmes d'optimisations.
- L'utilisation de notre noyau quantique pour réaliser une approche d'appariement de formes 3D à l'aide de descripteur shape context 3D.

Bibliographie

- [Ari03] Nafiz Arica. *Shape : Representation, Description, Similarity and Recognition*. PhD thesis, The middle east technical university, 2003.
- [BAKS12] A Ben Ayed, Mustapha Kardouchi, and S Selouani. Rotation invariant fuzzy shape contexts based on eigenshapes and fourier transforms for efficient radiological image retrieval. In *Multimedia Computing and Systems (ICMCS), 2012 International Conference on*, pages 266–271. IEEE, 2012.
- [Bal81] Dana H Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern recognition*, 13(2) :111–122, 1981.
- [BBP00] Stefano Berretti, Alberto Del Bimbo, and Pietro Pala. Retrieval by shape similarity with perceptual distance and effective indexing. *IEEE TRANSACTIONS ON MULTIMEDIA*, 2(4) :225–239, 2000.
- [Ber10] Maroun Bercachi. *Algorithme évolutionnaire à états pour l’optimisation difficile*. PhD thesis, Université Nice Sophia Antipolis, 2010.
- [BF92] F.W. Byron and R.W. Fuller. *Mathematics of Classical and Quantum Physics*. Dover books on physics and chemistry. Dover Publications, 1992.
- [BFM97] Thomas Back, David B Fogel, and Zbigniew Michalewicz. *Handbook of evolutionary computation*. IOP Publishing Ltd., 1997.
- [Bla02] Alexandre Blais. *Algorithmes et architectures pour ordinateurs quantiques supraconducteurs*. Université de Sherbrooke, 2002.
- [BM09] A. Ben-Menahem. *Historical Encyclopedia of Natural and Mathematical Sciences*. Number v. 1 in Historical Encyclopedia of Natural and Mathematical Sciences. Springer, 2009.
- [BMM06] Serge Belongie, Greg Mori, and Jitendra Malik. Matching with shape contexts. In *Statistics and Analysis of Shapes*, pages 81–105. Springer, 2006.
- [BMP00] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape context : A new descriptor for shape matching and object recognition. In *NIPS*, volume 2, page 3, 2000.
- [BMP02] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(4) :509–522, 2002.

- [Boo89] Fred L. Bookstein. Principal warps : Thin-plate splines and the decomposition of deformations. *IEEE Transactions on pattern analysis and machine intelligence*, 11(6) :567–585, 1989.
- [Bro92] Lisa Gottesfeld Brown. A survey of image registration techniques. *ACM computing surveys (CSUR)*, 24(4) :325–376, 1992.
- [CA05] M Emre Celebi and Y Alp Aslandogan. A comparative study of three moment-based shape descriptors. In *Information Technology : Coding and Computing, 2005. ITCC 2005. International Conference on*, volume 1, pages 788–793. IEEE, 2005.
- [Che01] Houcine Chebli. *Analyse Hilbertienne*. UVT, 2001.
- [CoI87] International Electrotechnical Commission and International Commission on Illumination. *International Lighting Vocabulary* .: Publication (International Electrotechnical Commission). Bureau Central de la Commission Electrotechnique Internationale, 1987.
- [Cra85] Michael Lynn Cramer. A representation for the adaptive generation of simple sequential programs. In *Proceedings of the First International Conference on Genetic Algorithms*, pages 183–187, 1985.
- [CRP98] P.K. Chawdhry, R. Roy, and R.K. Pant. *Soft Computing in Engineering Design and Manufacturing*. Springer London, 1998.
- [D⁺00] David P DiVincenzo et al. The physical implementation of quantum computation. *arXiv preprint quant-ph/0002077*, 2000.
- [Dav04] E Roy Davies. *Machine vision : theory, algorithms, practicalities*. Elsevier, 2004.
- [DBP97] Alberto Del Bimbo and Pietro Pala. Visual image retrieval by elastic matching of user sketches. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(2) :121–132, 1997.
- [DG06] Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pages 233–240. ACM, 2006.
- [Dir58] Paul Dirac. *The Principles of Quantum Mechanics*. Oxford university press, 1958.
- [Dum02] Paul Dumais. Hypothèses calculatoires en cryptographie duantidue. 2002.
- [Dur04] Nicolas Durand. Algorithmes génétiques et autres outils d’optimisation appliqués à la gestion de trafic aérien. *HDR de l’Institut Polytechnique de Toulouse*, 2004.
- [Fey67] Richard Phillips Feynman. *The character of physical law*, volume 66. MIT press, 1967.

- [Fey82] Richard P Feynman. Simulating physics with computers. *International journal of theoretical physics*, 21(6) :467–488, 1982.
- [Fit07] Richard Fitzgerald. What really gives a quantum computer its power ? *Physics Today*, 53(1) :20–22, 2007.
- [GH88] David E Goldberg and John H Holland. Genetic algorithms and machine learning. *Machine learning*, 3(2) :95–99, 1988.
- [Gos85] Ardeshir Goshtasby. Description and discrimination of planar shapes using shape matrices. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6) :738–743, 1985.
- [Gre86] John J Grefenstette. Optimization of control parameters for genetic algorithms. *Systems, Man and Cybernetics, IEEE Transactions on*, 16(1) :122–128, 1986.
- [Gro96] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219. ACM, 1996.
- [Gro02] Frédéric Grosshans. *Communication et cryptographie quantiques avec des variables continues*. PhD thesis, Université Paris Sud - Paris XI, 2002.
- [GS07] Ana Irene Ramirez Galarza and José Seade. *Introduction to classical geometries*. Springer, 2007.
- [GW99] Yoram Gdalyahu and Daphna Weinshall. Flexible syntactic matching of curves and its application to automatic hierarchical classification of silhouettes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(12) :1312–1328, 1999.
- [HAR11] Hedi HARZALLAH. *Contribution à la localisation et à la reconnaissance d’objets dans les images*. PhD thesis, Université de Caen, 2011.
- [HK00] Kuk-Hyun Han and Jong-Hwan Kim. Genetic quantum algorithm and its application to combinatorial optimization problem. In *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*, volume 2, pages 1354–1360. IEEE, 2000.
- [HK02] Kuk-Hyun Han and Jong-Hwan Kim. Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. *Evolutionary Computation, IEEE Transactions on*, 6(6) :580–593, 2002.
- [Hol75] John H Holland. *Adaptation in natural and artificial systems : An introductory analysis with applications to biology, control, and artificial intelligence*. U Michigan Press, 1975.
- [Hu62] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *Information Theory, IRE Transactions on*, 8(2) :179–187, 1962.

- [JWG00] David Jacobs, Daphna Weinshall, and Yoram Gdalyahu. Class representation and image retrieval with non-metric distances. *IEEE Trans. PAMI*, 22(583-600) :1, 2000.
- [Koz92] John R Koza. *Genetic programming : on the programming of computers by means of natural selection*, volume 1. MIT press, 1992.
- [Lac07] Xavier Lacour. *Information Quantique par Passage Adiabatique : Portes Quantiques et Décohérence*. PhD thesis, Université de Bourgogne, 2007.
- [LB03] Michel Le Bellac. Introduction à l’informatique quantique. *Cours donné à l’Ecole Supérieure de Sciences Informatiques (ESSI)*, 2003.
- [Ler13] Yves Leroyer. *Introduction a l’information quantique*. ENSEIRB-MATMECA, 2013.
- [Lév04] Benjamin Lévi. *Simulation de systèmes quantiques sur un ordinateur quantique réaliste*. PhD thesis, Université Paris-Diderot-Paris VII, 2004.
- [LJ07] Haibin Ling and David W Jacobs. Shape classification using the inner-distance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(2) :286–299, 2007.
- [LLE00] Longin Jan Latecki, Rolf Lakamper, and T Eckhardt. Shape descriptors for non-rigid shapes with a single closed contour. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 1, pages 424–429. IEEE, 2000.
- [Lon98] Sven Loncaric. A survey of shape analysis techniques. *Pattern recognition*, 31(8) :983–1001, 1998.
- [LYL10] Haibin Ling, Xingwei Yang, and Longin Jan Latecki. Balancing deformability and discriminability for shape matching. In *Computer Vision–ECCV 2010*, pages 411–424. Springer, 2010.
- [M⁺65] Gordon E Moore et al. Cramming more components onto integrated circuits, 1965.
- [Mah36] Prasanta Chandra Mahalanobis. On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2 :49–55, 1936.
- [Mah06] Karima Mahdi. L’optimisation multiobjectif et l’informatique quantique. Master’s thesis, Université Constantine 1, 2006.
- [MAK96] Farzin Mokhtarian, Sadegh Abbasi, and Josef Kittler. Efficient and robust retrieval by shape content through curvature scale space. In *Image Databases and Multi-Media Search*, pages 35–42, 1996.
- [MAO06] Julio Madera, Enrique Alba, and Alberto Ochoa. A parallel island model for estimation of distribution algorithms. In *Towards a New Evolutionary Computation*, pages 159–186. Springer, 2006.

- [Mit98] Melanie Mitchell. *An introduction to genetic algorithms*. MIT press, 1998.
- [MKJ08] Yang Mingqiang, Kpalma Kidiyo, and Ronsin Joseph. Shape matching and object recognition using chord contexts. In *Visualisation, 2008 International Conference*, pages 63–69. IEEE, 2008.
- [MMF14] Khalil M Mezghiche, Kamal E Melkemi, and Sebti Foufou. Matching with quantum genetic algorithm and shape contexts. In *(Accepted) IEEE/ACS 11th International Conference on Computer Systems and Applications (AICCSA)*, 2014.
- [MO99] Karsten Müller and Jens-Rainer Ohm. Contour description using wavelets. In *Workshop on Image Analysis for Multimedia Interactive Services*. Citeseer, 1999.
- [MT90] Silvano Martello and Paolo Toth. *Knapsack problems*. Wiley New York, 1990.
- [NC10] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.
- [NM96] Ajit Narayanan and Mark Moore. Quantum-inspired genetic algorithms. In *Evolutionary Computation, 1996., Proceedings of IEEE International Conference on*, pages 61–66. IEEE, 1996.
- [NNM96] Shree K Nayar, Sammeer A Nene, and Hiroshi Murase. Columbia object image library (coil 100). *Department of Comp. Science, Columbia University, Tech. Rep. CUCS-006-96*, 1996.
- [PI97] M Peura and J Iivarinen. Efficiency of simple shape descriptors. In *In Aspects of Visual Form*, pages 443–451. World Scientific, 1997.
- [Pou02] David Poulin. *Classicalité du calcul quantique*. PhD thesis, Université de Montréal, 2002.
- [Pre98] John Preskill. Lecture notes for physics 229 : Quantum information and computation. *California Institute of Technology*, 1998.
- [Ram13] Chafika Ramdane. *Apports du calcul quantique et des concepts possibilistes à la classification non supervisée évolutionnaire*. PhD thesis, Université Constantine 1, 2013.
- [Rec65] Ingo Rechenberg. Cybernetic solution path of an experimental problem. 1965.
- [Ric05] Julien Ricard. *Indexation et recherche d’objets 3D à partir de requêtes 2D et 3D*. PhD thesis, Université Claude Bernard - Lyon 1, 2005.
- [RP00] Eleanor Rieffel and Wolfgang Polak. An introduction to quantum computing for non-physicists. *ACM Computing Surveys*, 32(3) :300–335, 2000.
- [Ruc96] William Rucklidge. *Efficient visual recognition using the Hausdorff distance*. Springer-Verlag New York, Inc., 1996.

- [SBB⁺99] Lee Spector, Howard Barnum, Herbert J Bernstein, Nikhil Swamy, et al. Finding a better-than-classical quantum and/or algorithm using genetic programming. In *Proceedings of the Congress on Evolutionary Computation*, volume 3, pages 2239–2246, 1999.
- [Sch95] Benjamin Schumacher. Quantum coding. *Physical Review A*, 51(4) :2738, 1995.
- [Sch99] Antoine Schoeb. Analyse et comparaison de protocoles de purification de l'intrication quantique. Master's thesis, Université de Montréal, 1999.
- [Sch05] Maximilian Schlosshauer. Decoherence, the measurement problem, and interpretations of quantum mechanics. *Reviews of Modern Physics*, 76(4) :1267, 2005.
- [SCTK98] Daniel Sharvit, Jacky Chan, Hüseyin Tek, and Benjamin B Kimia. Symmetry-based indexing of image databases. In *Content-Based Access of Image and Video Libraries, 1998. Proceedings. IEEE Workshop on*, pages 56–62. IEEE, 1998.
- [SdLV99] Lambert Schomaker, Edward de Leau, and Louis Vuurpijl. Using pen-based outlines for object-based annotation and image-based queries. In *Visual Information and Information Systems*, pages 585–592. Springer, 1999.
- [SHB14] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image processing, analysis, and machine vision*. Cengage Learning, 2014.
- [Sho94] Peter W Shor. Algorithms for quantum computation : discrete logarithms and factoring. In *Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on*, pages 124–134. IEEE, 1994.
- [Sho95] Peter W Shor. Scheme for reducing decoherence in quantum computer memory. *Physical review A*, 52(4) :R2493, 1995.
- [SKK04] Thomas B Sebastian, Philip N Klein, and Benjamin B Kimia. Recognition of shapes by editing their shock graphs. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(5) :550–571, 2004.
- [SP94a] M Srinivas and Lalit M Patnaik. Adaptive probabilities of crossover and mutation in genetic algorithms. *Systems, Man and Cybernetics, IEEE Transactions on*, 24(4) :656–667, 1994.
- [SP94b] Mandavilli Srinivas and Lalit M Patnaik. Genetic algorithms : A survey. *Computer*, 27(6) :17–26, 1994.
- [Spa04] Anne Spalanzani. *Algorithmes évolutionnaires pour l'étude de la robustesse des systèmes de reconnaissance automatique de la parole*. PhD thesis, Université Joseph Fourier, 2004.

- [SSS94] Ramamurti Shankar, Ravi Shankar, and Ramamurti Shankar. *Principles of quantum mechanics*, volume 233. Plenum Press New York, 1994.
- [Tal09] Hichem Talbi. *Algorithmes évolutionnaires quantiques pour le recalage et la segmentation multiobjectif d'images*. PhD thesis, Université Constantine 1, 2009.
- [TBD04] Hichem Talbi, Mohamed Batouche, and Amer Draa. A quantum-inspired genetic algorithm for multi-source affine image registration. In *Image Analysis and Recognition*, pages 147–154. Springer, 2004.
- [Ter06] Amel Terki. Analyse des performances des algorithmes génétiques utilisant différentes techniques d'évolution de la population. Master's thesis, Université Constantine 1, 2006.
- [TS89] Antoine Taza and Ching Y Suen. Discrimination of planar shapes using shape matrices. *Systems, Man and Cybernetics, IEEE Transactions on*, 19(5) :1281–1289, 1989.
- [Vel01] Remco C Veltkamp. Shape matching : Similarity measures and algorithms. In *Shape Modeling and Applications, SMI 2001 International Conference on.*, pages 188–197. IEEE, 2001.
- [VH⁺58] Léon Van Hove et al. Von neumann's contributions to quantum theory. *Bulletin of the American Mathematical Society*, 64(3) :95–99, 1958.
- [VH00] Remco C Veltkamp and Michiel Hagedoorn. Shape similarity measures, properties and constructions. In *Advances in visual information systems*, pages 467–476. Springer, 2000.
- [VH01] Remco C Veltkamp and Michiel Hagedoorn. *State of the art in shape matching*. Springer, 2001.
- [WBY⁺12] Junwei Wang, Xiang Bai, Xinge You, Wenyu Liu, and Longin Jan Latecki. Shape matching and classification using height functions. *Pattern Recognition Letters*, 33(2) :134–143, 2012.
- [WLT99] Yu-Ping Wang, Seng Luan Lee, and Kazuo Toraiichi. Multiscale curvature-based shape representation using b-spline wavelets. *Image Processing, IEEE Transactions on*, 8(11) :1586–1592, 1999.
- [XHS08] Jun Xie, Pheng-Ann Heng, and Mubarak Shah. Shape matching and modeling using skeletal context. *Pattern Recognition*, 41(5) :1756–1767, 2008.
- [Yan05] Su Yang. Symbol recognition via statistical integration of pixel-level constraint histograms : A new descriptor. *IEEE transactions on pattern analysis and machine intelligence*, 27(2) :278–281, 2005.
- [Yan08] Mingqiang Yang. *Shape-based feature extraction and similarity matching*. PhD thesis, INSA de Rennes, 2008.

-
- [YKR⁺08] Mingqiang Yang, Kidiyo Kpalma, Joseph Ronsin, et al. A survey of shape feature extraction techniques. *Pattern recognition*, pages 43–90, 2008.
- [YSB08] Ang Yang, Yin Shan, and Lam Thu Bui. *Success in evolutionary computation*, volume 92. Springer, 2008.
- [YW07] Su Yang and Yuanyuan Wang. Rotation invariant shape contexts based on feature-space fourier transformation. In *Fourth International Conference on Image and Graphics*, 2007.
- [Zha11] Gexiang Zhang. Quantum-inspired evolutionary algorithms : a survey and empirical study. *Journal of Heuristics*, 17(3) :303–351, 2011.
- [ZL⁺01] Dengsheng Zhang, Guojun Lu, et al. A comparative study on shape retrieval using fourier descriptors with different shape signatures. In *Proc. of international conference on intelligent multimedia and distance education (ICI-MADE01)*, pages 1–9, 2001.
- [ZL04] Dengsheng Zhang and Guojun Lu. Review of shape representation and description techniques. *Pattern recognition*, 37(1) :1–19, 2004.
- [ZP04] Titus B Zaharia and Francoise J Preteux. 3d versus 2d/3d shape descriptors : A comparative study. In *Electronic Imaging 2004*, pages 47–58. International Society for Optics and Photonics, 2004.
- [ZR07] Gexiang Zhang and Haina Rong. Real-observation quantum-inspired evolutionary algorithm for a class of numerical optimization problems. In *Computational Science–ICCS 2007*, pages 989–996. Springer, 2007.
- [Zur03] Wojciech H Zurek. Decoherence and the transition from quantum to classical–revisited. *arXiv preprint quant-ph/0306072*, 2003.