

UNIVERSITE KASDI MERBAH OUARGLA
Faculté des Nouvelles technologies d'informatique et télécommunication
Département d'électronique et Télécommunication



Mémoire
MASTER ACADEMIQUE
Domaine : Sciences et technologies
Filière : Automatique
Spécialité : Automatique et Systems

Présenté par :

Meghaouri Nassef

Bouras Anis

Tebib Islam

Thème:

***Algorithme Flou pour l'Association
de Points d'Intérêts Entre les
Images Chevauchés***

Devant le jury :

Mr : CHAA Mourad

MCA

Président

UKM Ouargla

Mr : LATI Abdelhai

MAB

Encadreur/rapporteur

UKM Ouargla

Mr : TIDJANI Zakaria

MAA

Co-Encadreur

UKM Ouargla

Mr : BECHKA Larbi

MAA

Examineur

UKM Ouargla

Année universitaire 2019/2020



Dédicace

J'ai tout le plaisir de dédier ce modeste travail :

*À mes chers parents et toute ma famille, pour leur soutien et l'encouragement
pendant toutes ces années d'étude.*

À tous mes collègues du département d'électronique et télécommunication.

À toutes les personnes qui me connaissent de près ou de loin,

À tous les responsables du département d'électronique et télécommunication.

Et à tous mes enseignants qui ont contribué à notre formation.

« MEGHAOURI Nassef »



Dédicace

A mes chers parents, pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études,

A mes chères frères et sœurs pour leurs encouragements permanents, et leur soutien moral, A toute ma famille pour leur soutien tout au long de mon parcours universitaire,

Que ce travail soit l'accomplissement de vos vœux tant allégués, et le fruit de votre soutien infailible,

Merci d'être toujours là pour moi.

« BOURAS Anis »

A decorative border of colorful balloons (blue, pink, orange) with black strings, arranged in a rectangular frame around the page.

Dédicace

Je tiens c'est avec grande plaisir que je dédie ce modeste travail: a l'être le plus cher de ma vie , ma mère a celui qui m'a fait de moi un Homme, mon père a mes chers frères et sœurs. a tous mes amis de promotion de deuxième master en automatique et système a tous les membres de ma famille

« TEBIB Islam »



Remerciements

Avant de présenter notre modeste travail, il nous apparaît opportun de présenter d'abord nos remerciements :

Avant tout, Nous remercions Allah, le tout puissant qui nous a donnés à la fois, le courage et la puissance pour pouvoir mener ce travail à terme.

Nous manifestons nos remerciements distingués à notre encadreur Mr. LATI Abdel haï pour tous ses efforts fournis, son aide immense, ses conseils précieux et son orientation pour atteindre nos objectifs.

Et aussi, nous tenons à remercier l'Université Kasdi Merbah Ouargla

Un très grand remerciement et reconnaissance destinés à tous qui nous ont énormément aidés pendant la réalisation de ce mémoire à travers leurs conseils pertinents.

Et nous remercions les membres de jurys qui ont accepté d'évaluer ce travail de fin d'études. Sans oublier, toute personne ayant contribué, de près ou de loin, à l'accomplissement de ce travail.

En espérant que ce modeste travail soit à la hauteur et reflète ce que nous avons pu acquérir pendant cette période de projet.

Merci à tous



SOMMAIRE

SOMMAIRE

DEDICACE

REMERCIEMENT

SOMMAIRE.....	I
LISTE D'ABBREVIATIONS	III
LISTE DE FIGURES.....	IV
LIST DE TABLEAUX	V
INTRODUCTION GENERALE	1
CHAPITRE I:Etat de l'art de Techniques d'association de Points d'interets.....	3
I.1. Introduction	3
I.2. Caractéristiques d'Image	3
I.2.1. Pixel.....	3
I.2.3. Résolution.....	4
I.2.4. La luminance	4
I.2.5. L'histogramme	4
I.2.6. Le bruit	5
I.3. Types des images.....	5
I.3.1. L'image monochrome.....	5
I.3.2. L'image trichrome	5
I.4. Zones d'intérêts dans l'image	7
I.4.1. Les contours.....	5
I.4.2. Les points.....	6
I.4.3. Les regions.....	6
I.5. Détection des zones d'intérêts	7
I.6. Les points d'intérêts.....	8
I.6.1. Les avantages des points d'intérêts	8
I.6.2. La détection des points d'intérêts	8
I.6.3. Les Techniques de détection des points d'intérêts	8
I.7. L'association des points d'intérêts	9
I.7.1. Méthodes de Correlations	10
I.7.2. Méthodes de Descripteurs	10
I.7.3. Méthodes de Poursuite.....	11
I.7. Conclusion.....	12

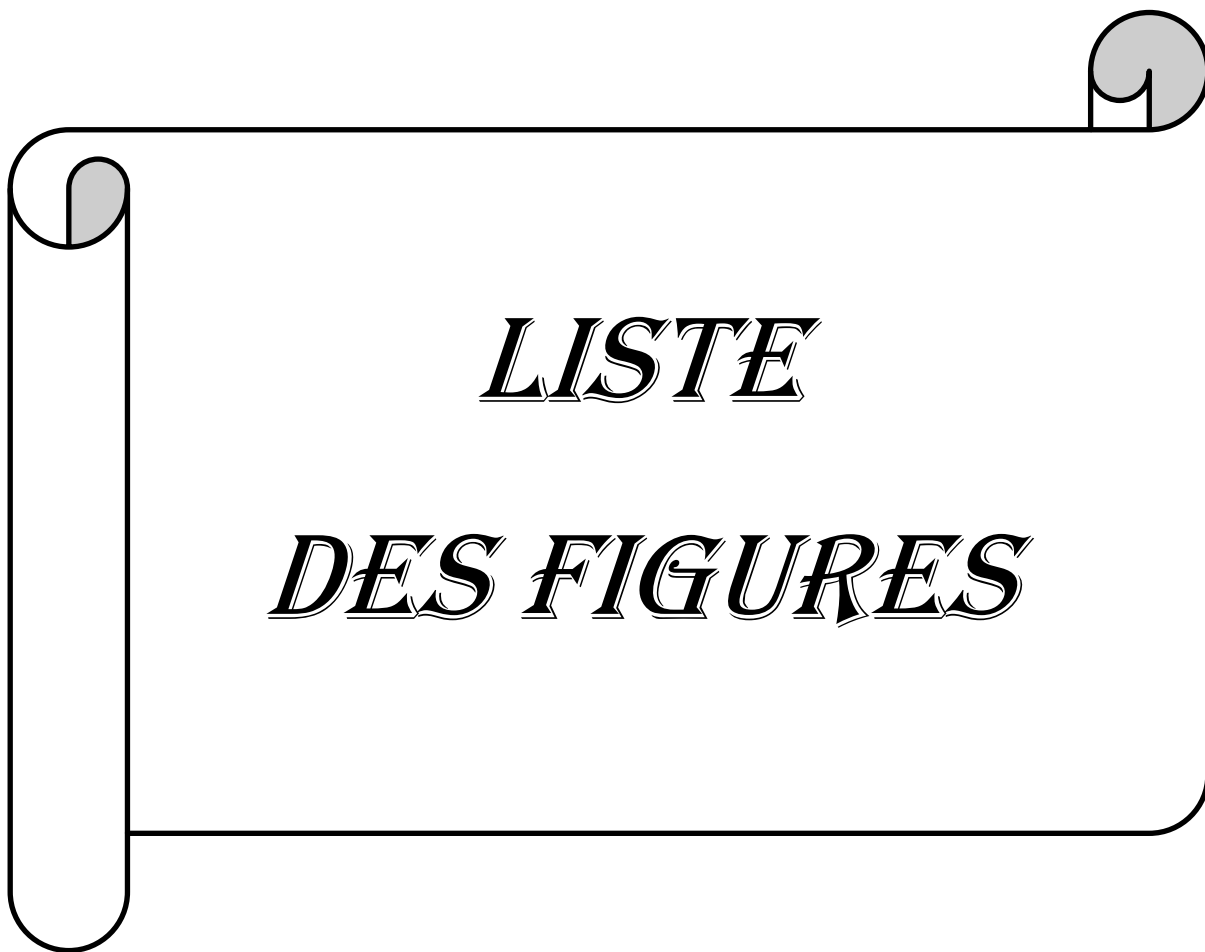
CHAPTRE II:Algorithms Floue d’Associations de Points d’interets	13
II.1. Introduction	13
II.2. Généralités sur la Logique Floue	13
II.2.1. Les ensembles Floues.....	13
II.2.2. Les Fonctions d’appartenace	13
II.2.3. Les Regles Flous	14
II.2.4. Les variable Flous	14
II.2.5. Système d’inférence	14
II.3. L’algorithme Proposé.....	15
II.3.1. Algorithme SURF	16
II.3.2. Algorithme FREAK	20
II.3.3. Système d’inférence	22
II.4. Conclusion.....	24
CHAPTRE III:Resultats de Simulation	25
III.1. Introduction.....	25
III.2. Environnement de développement.....	26
III.3.1. Environnement matériel	26
III.3.1. Environnement logiciel	25
III.3. Implementation de notre algorithme flou	26
III.3.1. Base de données	26
III.3.2. Détecteur de FAST	27
III.3.3. Association de points d’intérêts par l’algorithme flou.....	28
III.3.4. Les résultats obtenus	29
III.3.5. Discussion des resultats	33
3III.6. Conclusion	34
Conclusion Générale	36
References	38
Résumé	



*LISTE
D'ABBREVIATIONS*

Liste d'Abbreviations

SIFT	<i>Scale Invariant Features Transform</i>
Dpi	<i>Dots per inch</i>
PPI	<i>Points per inch</i>
RVB	<i>rouge, vert, bleu</i>
CMJN	<i>Cyan, Magenta, Jaune, Noire.</i>
TLS	<i>teinte, luminance, saturation.</i>
PI	<i>point d'intérêt</i>
RI	<i>Région d'intérêt</i>
ZI	<i>Zone d'intérêt</i>
SIFT	<i>Scale invariant feature transform</i>
SURF	<i>speed up robust feature</i>
SAD	<i>sum of absolute differences</i>
NCC	<i>normalized cross correlation</i>
SSD	<i>sum of squared differences</i>
KLT	<i>Kanade-Lucas-Tomasi</i>
FREAK	<i>Fast retina keypoints</i>
FAST	<i>Features from Accelerated Segment Test</i>
BRIEF	<i>Binary Robust Independent Elementary Features</i>
ORB	<i>Oriented FAST and Rotated BRIEF</i>
BRISK	<i>Binary Robust Invariant Scalable Keypoints</i>
MATLAB	<i>Matrix Laboratory</i>
ROI	<i>Regions Of Interest</i>



LISTE

DES FIGURES

Liste des Figures

Figure	Page
<i>Figure I-1: Pixel dans une image numérique.</i>	4
<i>Figure I-2: Image avec histogramme.</i>	4
<i>Figure I-3: Détection des points d'intérêts.</i>	6
<i>Figure I-4: Différents types de point d'intérêts.</i>	8
<i>Figure I-5: Recherche des points clé candidats dans l'image par corrélation</i>	10
<i>Figure II-1: Composants d'un système d'inférence floue.</i>	15
<i>Figure II-2: Dérivées partielles du second ordre gaussien dans les directions y et xy.</i>	17
<i>Figure II-3: Représentation de l'échelle de l'espace.</i>	18
<i>Figure II-4: Affectation d'orientation pour le descripteur de SURF.</i>	19
<i>Figure II-5: Construire du descripteur de SURF.</i>	19
<i>Figure II-6: Illustrations de la distribution des cellules ganglionnaires sur la rétine.</i>	20
<i>Figure II-7. Descripteur de FREAK</i>	22
<i>Figure II-8 : Schéma fonctionnel d'association à base de logique floue.</i>	23
<i>Figure III-1 : L'organigramme de notre technique proposé.</i>	25
<i>Figure III-2 : Test de segment appliqué sur un rayon de 3 pixels autour du pixel p.</i>	27
<i>Figure III-3 : Les étapes de notre algorithme pour trouver les associations correctes.</i>	28
<i>Figure III-4 : Les images chevauchées utilisées de la première scène.</i>	29
<i>Figure III-5 : Points d'intérêt détectés de la première scène à l'aide du détecteur FAST.</i>	29
<i>Figure III-6 : Points clés associés entre les images par l'algorithme flou de 1^{ère} scène.</i>	29
<i>Figure III-7 : Les images chevauchées utilisées de la deuxième scène.</i>	30
<i>Figure III-8 : Points d'intérêt détectés de la deuxième scène par détecteur FAST.</i>	30
<i>Figure III-9 : Points clés associés entre les images par l'algorithme flou de 2^{ème} scène.</i>	30
<i>Figure III-10 : Les images chevauchées utilisées de la troisième scène.</i>	31
<i>Figure III-11 : Points d'intérêt détectés de la troisième scène par du détecteur FAST.</i>	31
<i>Figure III-12 : Points clés associés entre les images par l'algorithme flou de 3^{ème} scène.</i>	31
<i>Figure III-13 : Les images chevauchées utilisées de la quatrième scène.</i>	32
<i>Figure III-14 : Points clés détectés de la quatrième scène à l'aide du détecteur FAST.</i>	32
<i>Figure III-15 : Points clés associés entre les images par l'algorithme flou de 4^{ème} scène.</i>	32



LISTE

DES TABLEAUX

Liste Des Tableaux

Tableau	Page
<i>Tableau III-1: L'effet du seuil FAST sur les primitives détectées.</i>	33
<i>Tableau III-2: Calcul de répétabilité entre les images chevauchées.</i>	34
<i>Tableau III-3: Calcul de pourcentage de correspondance correcte.</i>	34



INTRODUCTION

GENERALE

Introduction Générale

Les systèmes de vision par ordinateur visent à reproduire la vision humaine à l'aide d'ordinateurs. De tels systèmes peuvent aider les ordinateurs à apprendre et à prendre des mesures basées sur des données visuelles et peuvent trouver des modèles dans les images. De plus, ces systèmes peuvent repérer et trier différents types d'objets. La vision par ordinateur prend des images et des vidéos en entrée et donne des informations telles que la taille, la forme et la couleur en sortie. Ces systèmes utilisent des caméras et des capteurs pour collecter les entrées. Avec ces appareils, la vision par ordinateur peut étudier des images et des vidéos pour produire des résultats en temps réel.

Le traitement d'image applique des fonctions mathématiques aux images. Le résultat du traitement d'image peut fournir des données détaillées. Par conséquent, le traitement d'image effectue des opérations telles que le lissage, le filtrage et le contraste sur une image. Les ordinateurs perçoivent les images comme des signaux 2D constitués de lignes et de colonnes de pixels. L'entrée prise par les systèmes de traitement d'image peut fournir des données utiles dans de nombreuses applications. Par exemple, les hôpitaux utilisent le traitement d'image dans les processus d'imagerie biomédicale tels que la tomodensitométrie, l'échographie et l'IRM. Avec ces derniers, les médecins obtiennent des données importantes sur leurs patients. Ces données aident les médecins à confirmer leur diagnostic et à décider d'un traitement approprié. En outre, le traitement d'image est utilisé dans chaque smartphone. Après avoir cliqué sur une image, les smartphones utilisent des algorithmes de traitement d'image pour générer une image visuellement attrayante. Ici, les algorithmes ajustent l'exposition, le contraste, les hautes lumières, les ombres et de nombreuses autres propriétés d'une image.

Au cours des dernières décennies, les détecteurs et descripteurs de caractéristiques d'image sont devenus des outils populaires dans la communauté de la vision par ordinateur et ils sont largement appliqués dans un grand nombre d'applications. Représentation d'image, classification et récupération d'images, reconnaissance et mise en correspondance d'objets, reconstruction de scènes 3D, suivi de mouvement, classification de texture, les systèmes de localisation de robot et biométrie reposent tous sur la présence de caractéristiques stables et représentatives dans l'image. Ainsi, la détection et l'extraction des caractéristiques de l'image sont des étapes vitales pour ces applications.

La mise en correspondance des caractéristiques est une méthode efficace pour détecter une primitive spécifiée dans une région chevauchée dans des images de scène. Cette méthode détecte des objets uniques plutôt que plusieurs objets. L'algorithme est basé sur la comparaison et l'analyse des correspondances de points entre l'image de référence et l'image cible. Généralement, les performances des méthodes de correspondance sont basées sur les points d'intérêt et les méthodes d'association utilisés. La plupart des algorithmes d'appariement de caractéristiques sont basés sur la description de l'environnement des pintes clés détectées, puis sur l'application de certaines mesures de similitude et la prise de décisions. Mais dans certains cas; ces algorithmes donnent de mauvais résultats en raison de la qualité de l'image ou de l'incertitude de l'algorithme.

Dans de nombreuses applications de traitement d'image, des connaissances spécialisées doivent être utilisées pour des applications telles que la reconnaissance d'objets et l'analyse de scènes, La théorie des ensembles flous et la logique floue fournissent des outils puissants pour représenter et traiter les connaissances humaines sous forme de règles si-alors floues. De nombreuses difficultés de traitement d'image surviennent parce que les données, les tâches et les résultats sont incertains. Cette incertitude, cependant, n'est pas toujours due au caractère aléatoire mais à l'ambiguïté et à l'imprécision inhérentes aux données d'image. Donc; nous proposons d'utiliser deux techniques de description de caractéristiques importantes avec un système d'inférence floue pour identifier les associations correctes entre les images.

Notre mémoire est organisée en trois chapitres; dans le premier chapitre, nous avons donné une introduction aux images numériques et à leurs propriétés, ainsi que quelques techniques utilisées dans les littératures pour la détection et la correspondance des caractéristiques. Dans le deuxième chapitre, nous introduisons l'algorithme d'appariement flou proposé en définissant l'entrée et la sortie utilisées et les règles floues. Dans le troisième chapitre, nous présentons les résultats de simulation de notre algorithme et nous avons discuté les résultats obtenus sur la base de certaines mesures de performance. Enfin nous nous retrouvons avec une conclusion et quelques recommandations.



CHAPITRE I :

*ETAT DE L'ART SUR
L'ASSOCIATION DES
POINTS D'INTÉRÊTS*

I. Etat de l'art sur l'association de points d'intérêts

1. Introduction

L'image est une représentation d'une personne ou d'un objet par la peinture, la sculpture, le dessin, la photographie, le film, etc. C'est aussi un ensemble structure d'informations qui, après affichage sur l'écran, a une signification pour l'œil humain. Le terme d'image numérique désigne, dans son sens le plus général, toute image qui a été acquise, traitée et sauvegardée sous une forme codée représentable par valeurs numériques (une série de bits). L'image numérique est l'image dont la surface est divisée en éléments de tailles fixes appelés cellules ou pixels, ayant chacun comme caractéristique un niveau de gris ou couleurs prélevé à l'emplacement correspondant dans l'image réelle, ou calculé à partir d'une description interne de la scène à représenter [1-2].

L'extraction et l'association des points d'intérêts d'image est un lien technique très important dans le traitement d'image. Après des années de recherche approfondie, l'algorithme d'extraction et de description des points de caractéristique s'améliore et se perfectionne constamment. Il a un large éventail d'applications dans la correspondance d'images [1, 2], la récupération d'images, la reconnaissance d'images, le suivi de données vidéo, l'assemblage d'images, la classification d'images et bien d'autres aspects. Dans ce chapitre, nous entamerons une description générale sur les techniques d'association des points d'intérêt, nous allons parler des différentes méthodes d'association (corrélation, descripteur et poursuite). Ses étapes, selon les bases de celle-ci [3].

2. Les caractéristiques de l'image

L'image est un ensemble structuré d'informations caractérisé par les paramètres suivant :

2.1. Pixel

Pixel est le plus petit point de l'image, c'est une entité calculable qui peut recevoir une structure et une quantification. Si le bit est la plus petite unité d'information que peut traiter un ordinateur, le pixel est le plus petit élément que peuvent manipuler les matériels et logiciels d'affichage ou d'impression [2]. Voici (Figure I-1).

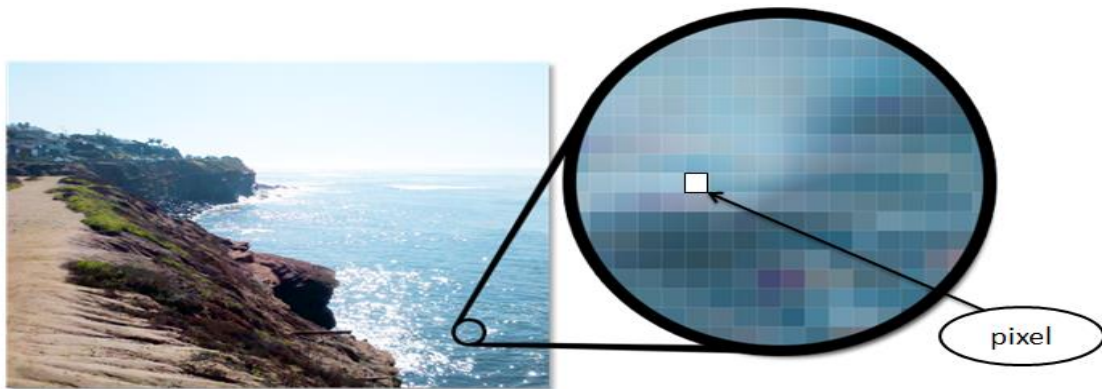


Figure I-1: Pixel dans une image numérique.

2.2. Résolution

Résolution est un nombre de pixels par pouce. Elle s'exprime en dpi (Dots Per Inch), en PPI (Points Per Inch) ou, pour les français, en ppp (Points Par Pouce). Ces trois unités sont équivalentes. Sachant qu'un pouce est égal à 2,54cm, on pourrait facilement calculer une résolution en pixels par centimètre mais ce n'est pas dans les habitudes [1].

2.3. La luminance

La luminance est le degré de luminosité des points de l'image. Elle est définie aussi comme étant le quotient de l'intensité lumineuse d'une surface par l'aire apparente de cette surface. Pour un observateur lointain, le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet [2].

2.4. Histogramme

L'histogramme des niveaux de gris ou des couleurs d'une image est une fonction qui donne la fréquence d'apparition de chaque niveau de gris (couleur) dans l'image. Pour diminuer l'erreur de quantification, pour comparer deux images obtenues sous des éclairages différents, ou encore pour mesurer certaines propriétés sur une image [2].

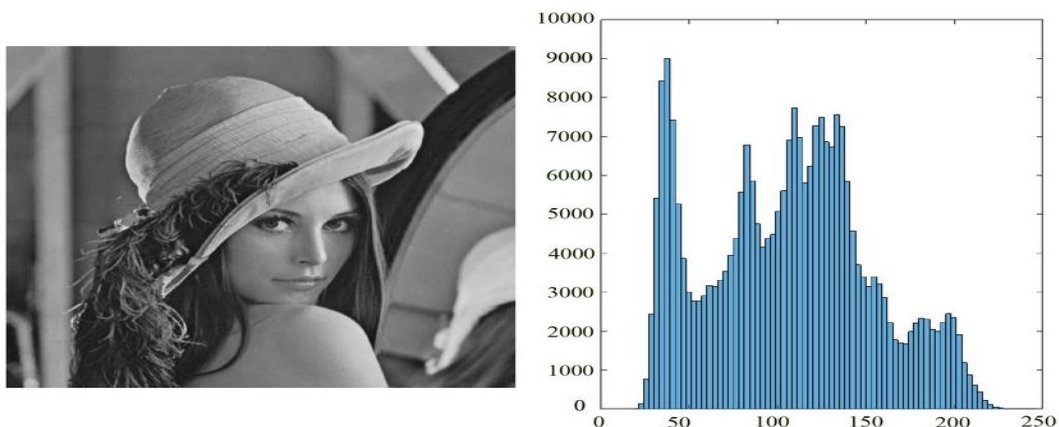


Figure I-2: Image avec histogramme.

2.5. Le bruit

Le bruit (ou grain) est la présence d'informations parasites qui s'ajoutent de façon aléatoire aux détails de la scène photographiée numériquement [3].

1. Types des images

1.1. Les images monochromes

Le terme monochrome signifie au sens littéral : « qui est d'une seule couleur ». En effet, le mot vient du grec : mono qui signifie « seul », et chroma, la couleur. Dans le contexte des images monochromes noir et blanc, qui nous intéresse plus particulièrement, les valeurs d'intensités associées aux pixels sont appelées niveaux de gris ou bien « nuancé de gris ». Le niveau de gris d'un pixel représente une mesure de l'accumulation de la lumière collectée dans la région du plan occupée par ce pixel [1].

1.2. Les images trichromes

C'est un système de codage de la couleur, couplé à celui de la représentation des niveaux de luminosité, doit être employé pour représenter les informations concernant la colorimétrie. De nombreux systèmes de codage ayant leurs particularités propres existent, comme par exemple [4] :

- RVB (rouge, vert, bleu).
- CMJN (Cyan, Magenta, Jaune, Noire).
- TLS (teinte, luminance, saturation).

2. Zones d'intérêts dans l'image

En vision par ordinateur et en traitement d'images, la détection de zones d'intérêt d'une image numérique consiste à mettre en évidence des zones de cette image jugées « intéressantes » pour l'analyse, c'est-à-dire présentant des propriétés locales remarquables. Il existe différents types de zones d'intérêts, et elles peuvent être classées comme suit.

2.1. Les Contours

Les premières méthodes proposées se fondent sur l'analyse des contours et des arêtes, donc c'est des zones où la luminance de l'image change brusquement, et comporte une discontinuité. La détection des contours dans une image réduit de manière significative la quantité de données en conservant des informations qu'on peut juger plus pertinentes [5]

2.2. Les Points

Les algorithmes de détection de points d'intérêts focalisent en général sur des points particuliers des contours, sélectionnés selon un critère précis.

Ainsi, les coins sont les points de l'image où le contour change brutalement de direction, comme aux quatre sommets d'un rectangle. Il s'agit de points particulièrement stables, et donc intéressants pour la répétabilité [6].

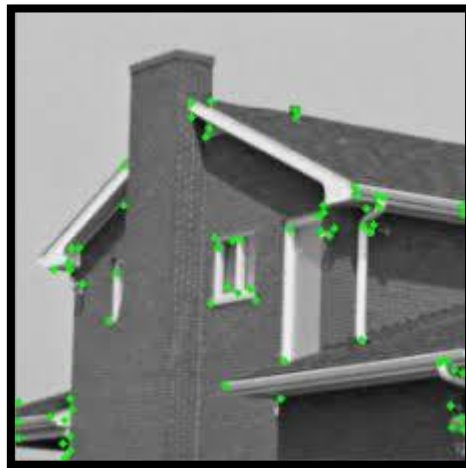


Figure I-3: Détection des points d'intérêts [6].

2.3. Les Régions

Il s'agit de zones d'intérêt plus générales que les points, utiles lorsque les structures recherchées dans une image ne correspondent pas à des points saillants : par exemple, lorsque l'image a subi un lissage important ou lorsque les contours sont épais et progressifs. Il est parfois intéressant de traiter une seule sous-région d'une image, en laissant les autres régions inchangées. C'est ce que l'on appelle communément le traitement des régions d'intérêt. Les sous-régions d'une image peuvent être spécifiées de manière pratique en utilisant des primitives graphiques mathématiques, telles que le point, la ligne, le cercle, le polygone, ou simplement une liste de positions de sommets.

On peut définir plusieurs régions d'intérêts (ROI) dans une image, les régions peuvent être de nature géographique telle que des polygones qui englobent des pixels contigus, ou elles peuvent être définies par une plage d'intensité (les pixels ne sont pas nécessairement contigus). Ces techniques commencent par identifier des points d'intérêt qui vont s'apparaitre être des sortes de centres des régions recherchées [7].

3. Détection des Zones d'intérêts

En vision par ordinateur et en traitement d'image, la détection de zone d'intérêt d'une image numérique consiste à mettre en évidence des zones de cette image jugées pertinentes pour l'analyse, c'est-à-dire présentant des propriétés locales remarquables. De telles zones peuvent apparaître, selon la méthode utilisée, sous la forme de points, ou de régions connexes rectangulaires ou non et qui constituent le résultat de la détection [8].

Le but de détecter la zone d'intérêt est de fournir une représentation qui permet de faire correspondre efficacement les structures locales entre images. En général, une zone d'intérêt possède les caractéristiques suivantes :

- Elle a une définition mathématique formelle,
- Elle a une position précise dans l'image,
- Elle est riche en informations visuelles locales,
- Elle est stable face à des variations locales et globales de l'image, i.e., elle conserve les mêmes informations visuelles en cas de variation.

On applique parfois, après la détection, un algorithme d'extraction de caractéristiques à une image. Cela consiste à calculer sur chaque zone détectée ce que l'on appelle un vecteur caractéristique qui résume le contenu de la zone, parmi les méthodes les plus répandues les histogrammes locaux (histogrammes de couleur, ou d'orientation des contours) et d'autres algorithmes (SIFT, SURF). Le vecteur caractéristique contient parfois des données issues directement de la détection, telles que l'orientation de l'arête ou la magnitude du gradient au point d'intérêt. Les vecteurs caractéristiques constituent une façon de décrire numériquement le contenu d'une image. De ce fait, ils sont souvent utilisés par des algorithmes plus globaux tels que la comparaison d'images [8].

4. Les Points d'Intérêts

Différentes primitives d'intérêt peuvent être distinguées : points, lignes, surfaces, volumes. Dans la plupart des cas il s'agit des points. Les termes employés peuvent être : Points d'intérêt, points clés, points dominants, points caractéristiques, points anatomiques. Les points d'intérêts, dans une image, correspondent à des doubles discontinuités de la fonction d'intensité. Celles-ci peuvent être provoquées, comme pour les contours, par des discontinuités de la fonction des discontinuités de profondeur. Un point d'intérêt est un point d'une image qui diffère de ses voisins immédiats, les coins sont les points de l'image où le contour change brutalement de direction, comme par exemple aux quatre sommets d'un rectangle. Il s'agit de points particulièrement stables, et donc intéressants [9-10].

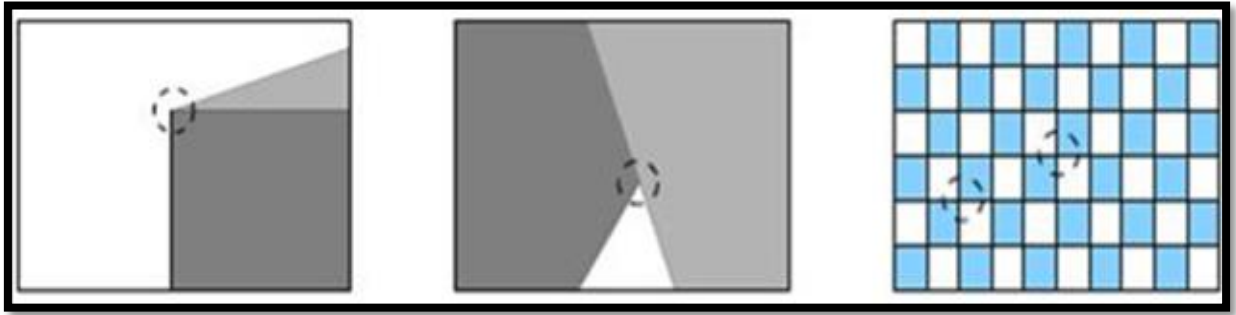


Figure I-4: Différents types de point d'intérêts.

4.1. Les Avantages des points d'intérêts

La détection de point d'intérêt dans les images est de plus utilisée afin de faciliter de nombreuses tâches, elle est des avantages en suit [10]:

- Sources d'informations plus fiables que les contours car plus de contraintes sur la fonction d'intensité.
- Robuste aux occultations (soit occulté complètement, soit visible)
- Présents dans une grande majorité d'images (contours).
- Différentes approches De nombreuses méthodes ont été proposées pour détecter des points d'intérêts.

4.2. La détection des points d'intérêts

En termes d'applications, l'utilisation de la détection des points d'intérêts est très vaste. Aujourd'hui, une application principale des points d'intérêt est de signaler des points dans le domaine de l'image qui sont susceptibles d'être utiles pour la correspondance d'images et la reconnaissance d'objets basée sur la vue. À cet effet, plusieurs types de détecteurs de coin et de détecteurs de taches se sont révélés très utiles dans des applications pratiques. Les détecteurs de taches et les détecteurs de coin ont également été utilisés comme primitives pour la reconnaissance de texture, l'analyse de texture et pour la construction de modèles 3D à partir de plusieurs vues d'objets texturés [11]. Si l'on vise à établir une distinction entre les détecteurs de coin et les détecteurs de taches, cela peut souvent être fait en termes de leurs propriétés de localisation au niveau des structures de coin. Pour une structure de jonction dans le domaine de l'image qui correspond à une intersection de bords physiques dans le monde tridimensionnel, les propriétés de localisation d'un détecteur de coin seront dans la plupart des cas bien meilleures que les propriétés de localisation qui seraient obtenues à partir d'un détecteur de blob [12].

Par conséquent, les détecteurs de coin auront dans de nombreux cas des avantages par rapport aux détecteurs de taches en termes d'erreur de localisation plus petite. En termes de concepts, il existe également une relation étroite entre la notion de points d'intérêt et les détecteurs de faitage, qui sont souvent utilisés pour signaler la présence d'objets allongés. De plus, en ce qui concerne les caractéristiques qui s'étendent le long des courbes unidimensionnelles dans l'espace image, il y a la notion connexe de détecteurs de bord qui répondent à des exigences similaires en termes de définitions opérationnelles, d'étendue bien définie, de contenu d'informations localement élevé et de répétabilité [13].

4.3. Les Techniques de détection des points d'intérêts

De nombreuses méthodes ont été proposées pour détecter des points d'intérêt. Elles peuvent être classées grossièrement suivant trois catégories [6] :

- I. **Approches par contours** : L'idée est de détecter les contours dans une image dans un premier temps. Des points sont ensuite extraits le long des contours en considérant les points de courbures maximales ainsi que les intersections de contours.
- II. **Approches par intensité** : L'idée est cette fois-ci de regarder directement la fonction d'intensité dans les images pour en extraire directement les points de discontinuités.
- III. **Approches à base d'un modèle** : Les points d'intérêts sont identifiés dans l'image par la mise en correspondance de la fonction d'intensité avec un modèle théorique de cette fonction des points d'intérêt considérés inférieures utilisées en détection d'objet en vision primaire.

5. L'association des points d'intérêt

5.1. Méthodes de corrélation

Les algorithmes de corrélation sont plus faciles à appliquer par rapport à d'autres algorithmes de correspondance basés sur des fonctionnalités. Cette méthode nécessite une mesure de similitude pour trouver les correspondances de points entre les deux vues superposées d'une scène [14]. Chaque point-clé de pixel dans une image a beaucoup de candidats possibles dans l'autre image à examiner afin de déterminer le meilleur point-clé de pixel de correspondance

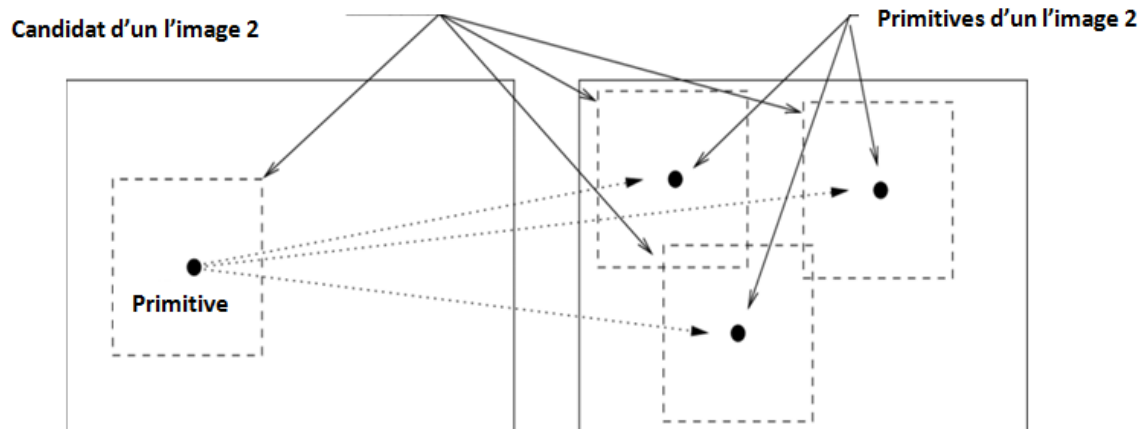


Figure I-5: Recherche des points clé candidats dans l'image par méthode de corrélation.

Plusieurs mesures de similitudes ont été proposées dans les littératures comme critères de correspondance comme la somme de la différence absolue (SAD), la somme de la différence carrée (SSD) et la corrélation croisée normalisée (NCC). Une étude expérimentale sur les différentes mesures de similitude en présence de diverses distorsions d'image a montré que le NCC fournit les meilleures performances. Le NCC est également plus robuste contre les changements d'éclairage que les SAD et SSD largement utilisés. Ainsi, le NCC est la mesure la plus appropriée pour déterminer efficacement l'emplacement d'un modèle dans une image en niveaux de gris. En raison de son efficacité, le NCC a été utilisé dans de nombreuses applications, cependant, en raison de son temps de calcul trop élevé [15].

Le problème associé à tous les algorithmes de correspondance de corrélation est que la taille des fenêtres de corrélation doit être précisément choisie. Si les fenêtres de corrélation sont trop petites, la variation d'intensité dans les fenêtres ne sera pas suffisamment distinctive et de nombreuses fausses correspondances peuvent se produire, cependant, si la fenêtre de corrélation est trop élevée, seules quelques correspondances peuvent être trouvées.

5.2. Méthodes de descripteur

Dans cette méthode, la meilleure correspondance candidate pour chaque point clé est trouvée en identifiant son voisin le plus proche dans la base de données de points clés à partir d'images. Les voisins les plus proches sont définis comme les points-clés avec une distance euclidienne minimale du vecteur descripteur donné. La probabilité qu'une correspondance soit correcte peut être déterminée en prenant le rapport de la distance du plus proche voisin à la distance du deuxième plus proche [16].

En utilisant l'algorithme du plus proche voisin, le but de similitude entre deux points clés est l'ampleur de la différence de leurs descripteurs, donc un score inférieur indique une correspondance plus étroite. Pour chaque fonctionnalité A dans l'image 1, nous calculons la différence entre A et chaque fonctionnalité B dans l'image 2. Nous acceptons une correspondance entre A et B si la différence entre eux est inférieure à la différence entre A et sa deuxième meilleure correspondance de l'image 2. En outre, pour éviter que les points de l'image 2 ne soient mis en correspondance avec plusieurs éléments de l'image 1, nous ne produisons que la meilleure correspondance pour chaque fonctionnalité de l'image 2 [16].

La distance euclidienne entre deux descripteurs peut être calculée par l'équation (I-1) :

$$d(A, B) = (\sum_i (A_i - B_i)^2)^{\frac{1}{2}} \quad (\text{I-1})$$

Cet algorithme donne des bons résultats si les caractéristiques de l'image sont bien extraites, et il est plus rapide que les méthodes basées sur la corrélation mais il est relativement insensible aux changements d'éclairage [17].

5.3. Méthode de poursuite

Cette méthode de mise en correspondance est largement utilisée pour différentes applications de traitement d'image telles que le recalage d'images, le suivi vidéo et la robotique, elle est basée sur la recherche de primitives dans un ensemble d'images qui sont superposées (séquence d'image), puis sur la mise en correspondance de ces points en recherchant un ensemble d'emplacements de points d'intérêt probables dans chaque images successives où la quantité de mouvement et la déformation de l'apparence devrait être faible. La première étape consiste à identifier et à suivre des cibles. Une cible est un point spécifique de l'image qu'un algorithme de poursuite. Le choix de ces cibles dépend de l'algorithme de poursuite, mais ce sont souvent des endroits lumineux/sombres, des arêtes ou des coins. L'important est que chaque cible représente un point d'intérêt de la surface d'un objet réel. Lorsqu'elle est suivie, une cible devient une suite de coordonnées bidimensionnelles représentant la position de la cible à travers la séquence d'image [18].

Parmi les algorithmes les plus utilisés dans ce domaine Kanade-Lucas-Tomasi (KLT) c'est un algorithme de poursuite (suivi vidéo) est basée sur la recherche de points dans un ensemble d'images qui sont superposées, cet algorithme a été suggéré par Tomasi et Kanade. Cette méthode est utilisée dans les systèmes de vision par ordinateur pour suivre les caractéristiques d'une série d'images. KLT repose sur la détection de points clés dans la première image, puis sur le suivi de ces fonctionnalités dans la séquence d'images [19].

6. Conclusion

Nous avons vu dans ce premier chapitre, brièvement, un état de l'art sur le développement des techniques de détection et l'extraction des points d'intérêts ainsi que la mise en correspondance des images grâce à ses points en utilisant différentes méthodes corrélation, descripteur et poursuite.

Dans le chapitre suivant, nous allons reprendre en détail deux algorithmes choisis pour effectuer l'association des points d'intérêt en utilisant la logique floue.



CHAPITRE II :
ASSOCIATION FLOUE
DES POINTS
D'INTÉRÊT

II. Association Floue des Points d'Intérêt

1. Introduction

Dans ce chapitre ; nous définirons les notions de base de la logique floue, L'idée de la logique floue est de «capturer» l'imprécision de la pensée humaine et l'exprimer avec des outils mathématiques appropriés. Puis nous allons analyser les algorithmes SURF, FREAK et les étapes d'extraction des points d'intérêts et calcul des descripteurs. Et comment réaliser un algorithme flou pour l'association des points d'intérêt.

2. Généralité sur La logique floue

La logique floue est un outil de l'intelligence artificielle, et est une extension de la logique booléenne créée par Lotfi Zadeh en 1965 [20]. Elle est formalisée par une théorie mathématique des ensembles flous, qui est une généralisation de la théorie des ensembles classiques. En introduisant la notion de degré dans la vérification d'une condition, permettant ainsi à une condition d'être dans un autre état que vrai ou faux (sont des réels entre 0 et 1), la logique floue confère une flexibilité très appréciable aux raisonnements qui l'utilisent, ce qui rend possible la prise en compte des imprécisions et des incertitudes [21].

2.1. Les sous-ensembles flous

La théorie des sous-ensembles flous est en fait selon Zadeh, c'est un pas vers un rapprochement entre la précision des mathématiques classiques et la subtile imprécision du monde réel. Qui est une généralisation de la théorie des ensembles classiques. Indifféremment les termes sous-ensembles flous et ensembles flous. Les ensembles classiques sont également appelés ensembles nets, par opposition à flou, et de même la logique classique est également appelée logique booléenne ou binaire. Les ensembles sont souvent représentés sous forme graphique [22].

2.2. Les fonctions d'appartenance

Un ensemble flou est défini par sa « fonction d'appartenance », Le concept d'appartenance est primordial dans la théorie des ensembles: il désigne le fait qu'un élément fasse partie ou non d'un ensemble. Par exemple, l'entier 7 appartient à l'ensemble {6; 7; 9}. Au contraire, l'entier 5 n'appartient pas à l'ensemble {6; 7; 9}.

Pour simplification, l'appartenance est symbolisée par le caractère \in et la non-appartenance par le même symbole, mais barré \notin . Ainsi, nous avons $7 \in \{6; 7; 9\}$ et $5 \notin \{6; 7; 9\}$. Une fonction d'appartenance également appelée fonction indicatrice ou encore fonction caractéristique est une fonction qui explicite l'appartenance ou non à un ensemble [23]. Le choix de la fonction d'appartenance est fait de façon arbitraire au fil des tests on choisit si c'est la forme sigmoïde, gaussienne, triangulaire, trapézoïdale, singleton.

2.3. Les règles floues

Une règle floue est de la forme : Si je rencontre telle situation Alors j'en tire telle conclusion. La situation appelée prémisse ou antécédent de la règle, est définie par une combinaison de relation de la forme x est A pour chacune des composante du vecteur d'entrée. La partie conclusion de la règle est appelée conséquence, ou encore simplement conclusion [24]. Zadeh a été le premier à introduire la notion de règle floue sous la forme :

❖ **Règle** : Si x est A , alors y est B .

Une règle floue (Si...Alors) est représentée par une implication floue ayant la même fonctionnalité que celle utilisée dans la logique classique. Par exemple, la règle floue: Si x_1 est A_1 et x_2 est A_2 alors y est B où A_1 , A_2 , et B sont des ensembles flous représentés respectivement par les fonctions d'appartenances (x_1) , (x_2) et (y) [25].

2.4. Les variables floues

A l'inverse de la logique classique où les variables binaires sont définies par deux états « vrai » ou « faux », Les variables floues supportent toute une gradation entre ces deux possibilités, En logique floue, les variables ne sont plus binaires mais présentent une multitude de valeurs possibles entre le « vrai » et le « Faux » [20].

2.5. Système d'inférence

Un Système d'Inférence Floue son rôle est de transformer les données d'entrée en données de sortie à partir de l'évaluation d'un ensemble des règles, Les entrées sont issues du processus de fuzzification et l'ensemble de règles sont fait par l'expert, les composants essentielles d'un système d'inférence floue sont les suivantes [26] :

- **Fuzzification** : Elle permet de transformer une valeur numérique de l'entrée en une partie floue définie sur un espace de représentation lié à l'entrée.

- **Inférence** : consiste à utiliser le moteur d'inférence, qui est un mécanisme permettant de condenser l'information d'un système à travers d'un ensemble de règles définies pour la représentation d'un problème.
- **Defuzzification** : Elle permet de transformer une partie floue de l'entrée en une valeur numérique définie sur un espace de représentation lié à l'entrée.

La figure II-2 montre la relation entre les différentes parties d'un système d'inférence floue :

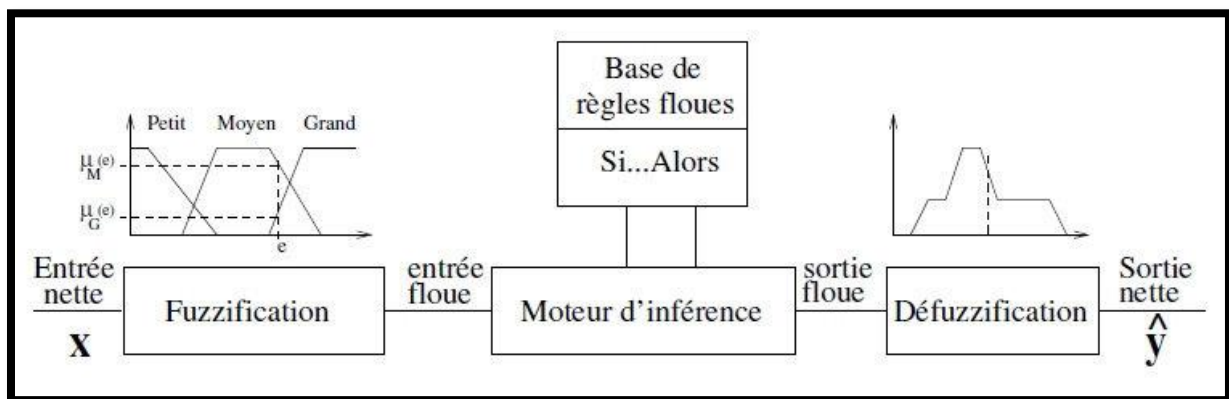


Figure II-1: Composants d'un système d'inférence floue.

3. L'Algorithme Proposé

Imprécision et incertitude sont inhérentes aux images, et trouvent leur origine à différents niveaux : phénomènes observés, capteurs et acquisitions de données, processus numériques de reconstruction, nature des images et modes de représentation de leurs éléments constitutifs, etc. La logique floue a été conçue pour déterminer le degré de similitude entre les images de test et de référence. Les entrées de logique floue pour chaque paire de points clés appariés (k_1, k_2) obtenus à l'aide de l'association SURF et FREAK sont les suivants:

1. Mesure de similarité obtenue à l'aide de l'association SURF standard.
2. Mesure de similarité obtenue à l'aide de l'association FREAK standard.
3. Appartenance du point clé (k_1) à sa région (r_1) dans le test image.
4. Appartenance du point clé (k_2) à sa région (r_2) dans image de référence.

L'appartenance d'un point d'intérêts dans r_1 est donnée par le rapport entre le nombre de points d'intérêts dans r_1 correspondant au nombre de points d'intérêts dans r_2 et le nombre de points d'intérêts dans r_1 . De même l'appartenance du point d'intérêts (k_2) à r_2 est également définie comme vice versa.

3.1. SURF algorithm

L'algorithme SURF (Speed Up Robust Features) ou bien en français caractéristiques robustes accélérées cet algorithme est inspiré par l'algorithme SIFT (Scale Invariant Feature Transform), c'est un algorithme de détection de caractéristique et un descripteur, été publié et présenté lors de la Conférence européenne de 2006 sur la vision par ordinateur. Est un algorithme rapide et robuste pour la représentation locale et la comparaison invariante des images. L'intérêt principal de l'approche SURF réside dans son calcul rapide, permettant ainsi des applications en temps réel telles que le suivi et la reconnaissance d'objets. [27] SURF est composé en deux étapes :

3.1.1. Détection de Points d'Intérêts

Pour détecter les points clés, nous pouvons utiliser plusieurs types de détecteurs de primitives; comme Harris, FAST ... etc L'approche de SURF pour la détection de points d'intérêt utilise une approximation de matrice Hessienne très basique.

i. L'intégrale d'image

L'intégrale est utilisée comme un moyen rapide et efficace de calculer la somme des valeurs (valeurs de pixels) dans une image donnée - ou un sous-ensemble rectangulaire d'une image. Il est principalement, utilisé pour calculer l'intensité moyenne d'une image donnée :

$$I(x) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(i, j) \quad (\text{II} - 1)$$

Cette méthode permet un calcul rapide des filtres à convolution de type boîte. L'entrée d'un intégrale d'image $I \sum(\mathbf{x})$ à un emplacement $\mathbf{x} = (\mathbf{x}, \mathbf{y})^T$ représente la somme de tous les pixels de l'image d'entrée \mathbf{I} dans une région rectangulaire formée par l'origine et \mathbf{x} [28].

ii. La matrice Hessienne pour la détection de points d'intérêts

La technique de SURF utilise la matrice de Hessienne en raison de ses bonnes performances en temps de calcul et en précision. Plutôt que d'utiliser une mesure différente pour sélectionner l'emplacement et l'échelle, le SURF repose sur le déterminant de la matrice Hessienne pour les deux. Étant donné un pixel, La matrice Hessienne de ce pixel est donnée par l'équation suivant :

$$H(x, y, \sigma) = \begin{bmatrix} l_{xx}(x, \sigma) & l_{xy}(x, \sigma) \\ l_{xy}(x, \sigma) & l_{yy}(x, \sigma) \end{bmatrix} \quad (\text{II-2})$$

Où $L_{xx}(x, \sigma)$ est le résultat de convolution de la dérivée de second ordre du filtre gaussien $\frac{\delta^2}{\delta x^2}g(\sigma)$ avec l'image I au point x , et de même pour $L_{xy}(x, \sigma)$ et $L_{yy}(x, \sigma)$.

Afin de calculer le déterminant de la matrice de Hessienne, nous devons d'abord appliquer une convolution avec un noyau gaussien, puis une dérivée de second ordre. Après le succès de Lowe avec les approximations LoG (SIFT), SURF pousse encore plus loin l'approximation (à la fois la convolution et la dérivée de second ordre) avec des filtres de boîte. Ces dérivées gaussiennes approximatives du second ordre peuvent être évaluées à un temps de calcul très faible en utilisant des intégrales d'images et indépendamment de la taille, et cela fait partie de la raison pour laquelle SURF est rapide.

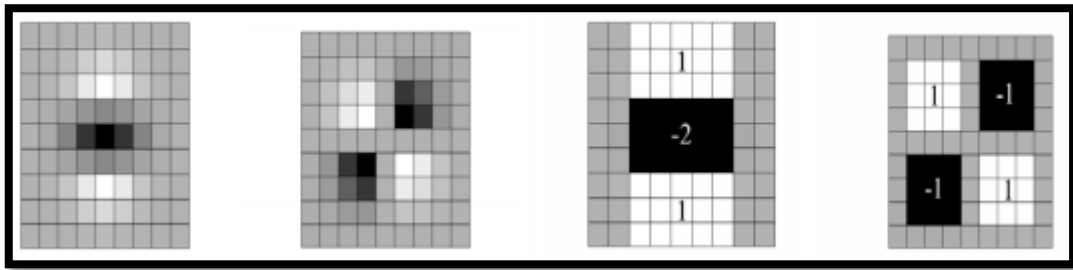


Figure II-2: Les dérivées partielles du second ordre gaussien dans les directions y et xy .

Les filtres de boîte 9×9 dans les images ci-dessus sont des approximations pour les dérivées gaussiennes de second ordre avec $\sigma = 1,2$. Nous désignons ces approximations par D_{xx} , D_{yy} et D_{xy} . Maintenant, nous pouvons représenter le déterminant de la Hessienne (approximé) comme (ou ω est constant) [29] :

$$\det(H_{app}) = D_{xx}D_{yy} - (\omega D_{xy})^2 \quad (\text{II-3})$$

iii. Représentation de l'échelle de l'espace

Les espaces d'échelle sont représentés par de pyramides d'images. Les images sont lissées à plusieurs reprises avec un gaussien et ensuite sous-échantillonnées afin d'atteindre un niveau plus élevé de la pyramide. En raison de l'utilisation de filtres rectangulaires et d'images intégrales, le SURF n'a pas à appliquer de manière itérative le même filtre à la sortie d'une couche précédemment filtrée, mais peut à la place appliquer de tels filtres de n'importe quelle taille à exactement la même vitesse directement sur l'image d'origine, et même en parallèle. Par conséquent, l'espace d'échelle est analysé en augmentant la taille du filtre ($9 \times 9 \rightarrow 15 \times 15 \rightarrow 21 \times 21 \rightarrow 27 \times 27$, etc.) plutôt que de réduire de manière itérative la taille de l'image.

Ainsi, pour chaque nouvelle octave, l'augmentation de la taille du filtre est doublée simultanément, les intervalles d'échantillonnage pour l'extraction des points d'intérêt (σ) peuvent également être doublés, ce qui permet la mise à l'échelle du filtre à coût constant. Afin de localiser les points d'intérêt dans l'image et sur des échelles, une suppression non maximale dans un voisinage $3 \times 3 \times 3$ (abscisse x ordonnée x échelles) est appliquée [30].

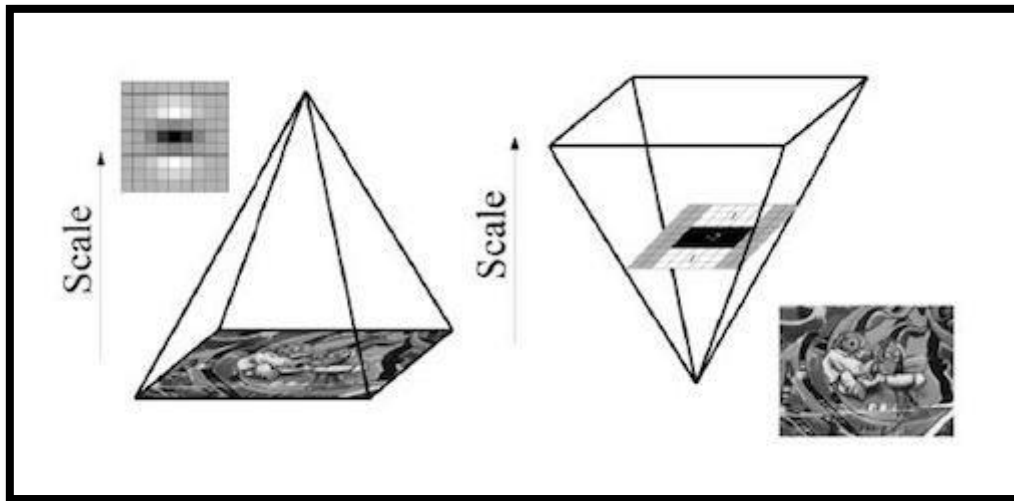


Figure II-3: Représentation de l'échelle de l'espace.

3.1.2. Description de Points d'Intérêts

La création des descripteurs de SURF se déroule en deux étapes. La première étape consiste à fixer une orientation reproductible à partir d'informations issues d'une zone circulaire autour du point clé. Ensuite, une région carrée alignée est construite sur l'orientation sélectionnée et le descripteur de SURF est créé [31] :

i. Affectation d'orientation

Afin d'être invariant à la rotation, la technique de SURF essaie d'identifier une orientation reproductible pour les points d'intérêt comme suivante :

- 1) SURF calcule d'abord les réponses des Haar-ondelettes dans les directions x et y, et ce dans un voisinage circulaire de rayon $6s$ autour du point clé, avec l'échelle « s » à laquelle le point clé a été détecté. De plus, l'étape d'échantillonnage est dépendante de l'échelle s, et les réponses en ondelettes sont calculées à cette échelle actuelle s. En conséquence, à des échelles élevées, la taille des ondelettes est grande. Par conséquent, les intégrales d'images sont à nouveau utilisées pour un filtrage rapide.

- 2) Ensuite, la somme des réponses verticales et horizontales est calculé en ondelettes dans une zone de numérisation, puis l'orientation de numérisation est modifié (ajoute de $\pi/3$) et recalculé jusqu'à ce que l'orientation avec la plus grande valeur de somme est trouvé, cette orientation est l'orientation principale de descripteur.

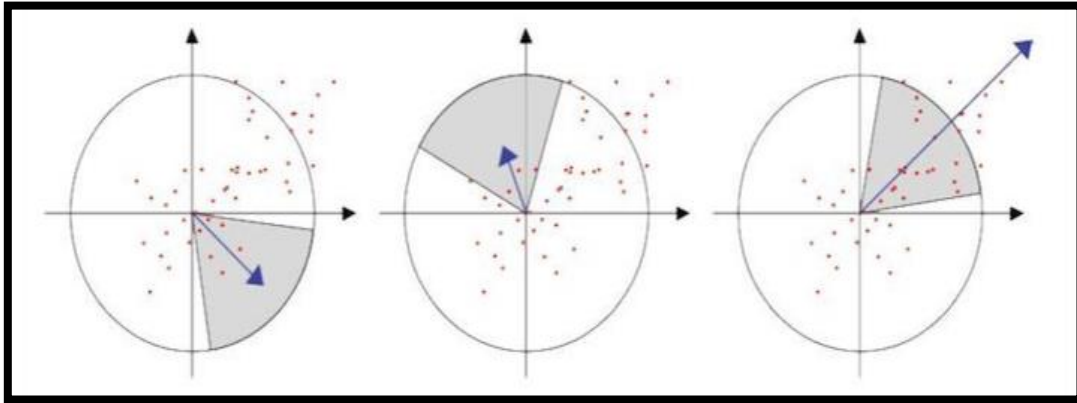


Figure II-4: Affectation d'orientation pour le descripteur de SURF.

ii. Composants du descripteur

- 1) La première étape consiste à construire une région carrée centrée autour du point clé et orientée selon l'orientation que nous avons déjà obtenue ci-dessus. La taille de cette fenêtre est de 20 s.
- 2) Ensuite, la région est régulièrement divisée en sous-régions carrées (4×4) plus petites. Pour chaque sous-région, quelques caractéristiques simples à 5×5 points d'échantillonnage sont régulièrement calculés espacés. Pour simplicité, les réponses en ondelettes de Haar dans le sens horizontal et dans le sens vertical (dx et dy) sont appliqués (taille de filtre 2s). Pour augmenter la robustesse aux déformations géométriques et aux erreurs de localisation, les réponses dx et dy sont d'abord pondérées avec une gaussienne ($\sigma = 3,3s$) centrée au point clé.

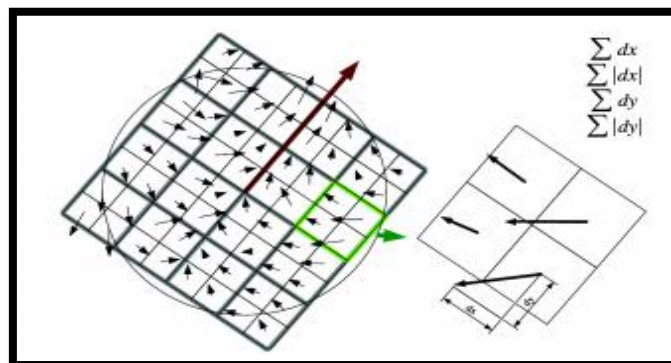


Figure II-5: Construire du descripteur de SURF.

3.2. L'algorithme de FREAK

Fast Retina Keypoint (FREAK) est une technique de description de de point-clé inspiré du système visuel humain et plus précisément de la rétine. Une cascade de chaînes binaires est calculée en comparant efficacement les intensités d'image sur un motif d'échantillonnage rétinien. Les expériences montrent que les FREAKs sont général plus rapide à calculer avec une charge de mémoire inférieure et aussi plus robuste que SIFT, SURF ou BRISK. De nombreuses grilles d'échantillonnage permettent de comparer des paires des intensités de pixels. BRIEF et ORB utilisent des paires aléatoires. BRISK utilise un motif circulaire où les points sont également espacés sur des cercles concentriques. Nous proposons d'utiliser la grille d'échantillonnage rétinienne qui est également circulaire à la différence d'avoir une densité de points plus élevée près du centre. La densité des points chute exponentiellement comme le montre la figure II-6 [32].

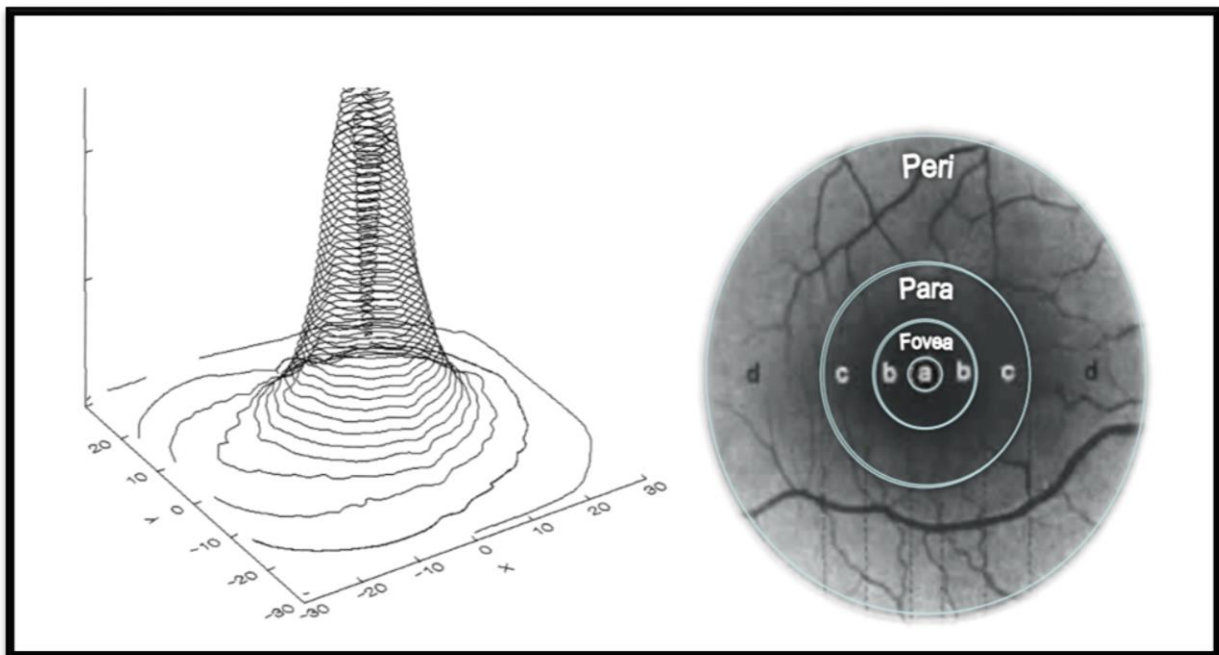


Figure II-6: Illustrations de la distribution des cellules ganglionnaires sur la rétine.

Chaque point d'échantillonnage doit être lissé pour être moins sensible au bruit. BRIEF et ORB utilisent le même noyau pour tous les points du patch. Pour correspondre au modèle de la rétine, nous utiliser des noyaux de tailles différentes pour tous les points d'échantillonnage similaires à BRISK. La différence avec BRISK est l'exponentiel changement de taille et les champs récepteurs qui se chevauchent.

Figure II-6 illustre la topologie des champs récepteurs. Chaque cercle représente les écarts types des noyaux gaussiens appliqués aux points d'échantillonnage correspondants. Il est observé expérimentalement que la modification de la taille des noyaux gaussiens par rapport à la rétine log-polaire modèle conduit à de meilleures performances. De plus, le chevauchement des champs récepteurs augmente également les performances. La raison possible est qu'avec le chevauchement présenté dans la figure4, plus d'informations sont capturées. Nous ajoutons une redondance apporte plus de pouvoir discriminant. Considérons les intensités mesurées aux champs récepteurs A, B et C où: $I_a > I_b, I_b > I_c, \text{ et } I_a > I_c$ [33].

Si les champs n'ont pas de chevauchement, le dernier test $I_a > I_c$ n'ajoute aucune information discriminante. Cependant, si les champs se chevauchent, des informations partiellement nouvelles peuvent être encodées. En général, l'ajout de redondance nous permet d'utiliser moins de champs récepteurs qui sont une stratégie connue employée dans la détection compressée ou l'apprentissage par dictionnaire. Une telle redondance existe également dans les champs récepteurs de la rétine [34].

3.2.1. Détection de Primitives de FREAK

Dans le cadre de notre étude, nous considérons que l'utilisation de FAST (Features From Accelerated Segment Test) pour la détection des primitives est un excellent moyen pour faire l'évaluation du FREAK. Ou nous pouvons utiliser autre détecteur de primitives

3.2.2. Description de Primitives de FREAK

Le descripteur FREAK est basé sur des comparaisons d'intensité entre différentes paires de points d'échantillonnage lissés [35]. En considérant une paire de points d'échantillonnage $P_a = (P_i, P_j)$, où $i, j \in \{1, 2, \dots, N\}$ et $i \neq j$, l'algorithme définit une comparaison d'intensité codée binaire $s(P_a)$ sur ce paire comme suit:

$$s(P_a) = \begin{cases} 1 & \text{if } P_i > P_j \\ 0 & \text{otherwise} \end{cases} \quad (II - 4)$$

La comparaison ci-dessus crée la base pour la construction du descripteur FREAK F contenant une chaîne de N bits:

$$F = \sum_{0 \leq a < N} 2^a s(P_a) \quad (II - 5)$$

FREAK spécifie 45 points d'échantillonnage pour l'orientation de descripteurs, comme le montre la figure II-7[36],

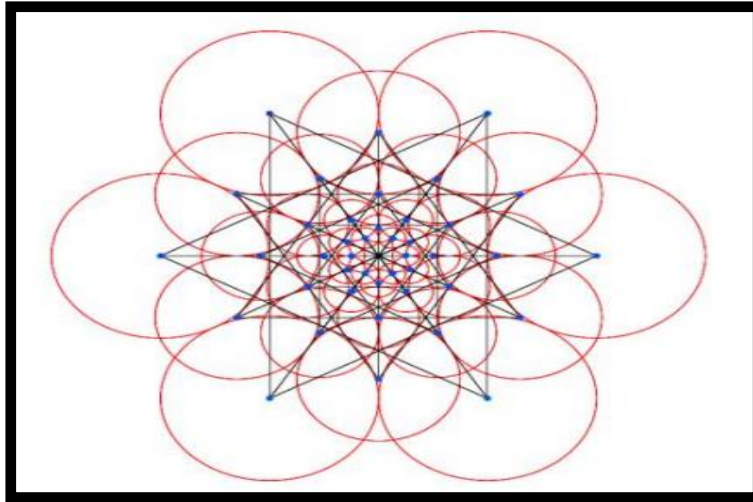


Figure II-7 : Descripteur de FREAK [36].

Lorsque les cercles bleus sont des points d'échantillonnage, les cercles rouges sont des champs récepteurs lissés et les lignes représentent des paires de points clés. Ce motif est symétrique et est utilisé pour générer le gradient local. L'orientation O d'un point clé donné peut être calculée à l'aide de l'équation suivante:

$$O = \frac{1}{M} \sum_{\substack{P_i, P_j \in G \\ i \neq j}} (P_i, P_j) \frac{T(P_i) - T(P_j)}{\|T(P_i) - T(P_j)\|} \quad (II - 6)$$

Où : G est l'ensemble de toutes les paires utilisées pour calculer les gradients locaux et M est le nombre de paires dans G .

$T(P_i)$ désigne une fonction renvoyant le vecteur 2D des coordonnées spatiales du centre du champ récepteur, c'est-à-dire le vecteur des coordonnées du k -ème point d'échantillonnage $T(P_i) = (x_i, y_i)$.

3.3. Système d'inférence

L'appartenance d'un point clé dans $r1$ est donnée par le rapport entre le nombre de points-clés dans $r1$ correspondant au nombre de points-clés dans $r2$ et le nombre de dans $r1$. De même l'appartenance du point clé ($k2$) à $r2$ est également définie comme vice versa. Le schéma synoptique de la correspondance floue est présenté dans Figure 4. Les détails de la logique floue sont décrits ci-dessous. Le schéma synoptique de la correspondance floue est présenté dans Figure 13. Les détails de la logique floue sont décrits ci-dessous.

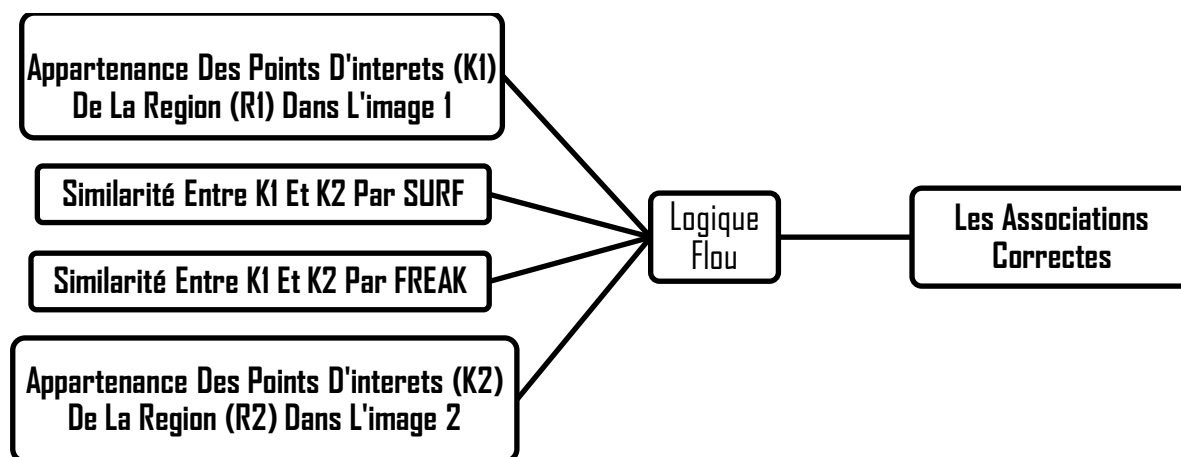


Figure II-8 : Schéma fonctionnel d'association à base de logique floue.

Les fonctions d'appartenance pour la similarité SURF et de FREAK des variables d'entrée mesurées sont définies par des fonctions gaussiennes pour 'faible', «Moyen» et «élevé» (voir la figure II-9 (a-b)). La variable d'entrée «Appartenance» est définie par la fonction sigmoïde, indiquée dans Figure II-9 (c). La variable de sortie est définie par des fonctions gaussiennes pour «très faible», «faible», «moyen», «élevé »et« très élevé »(voir la figure II-9(d)).

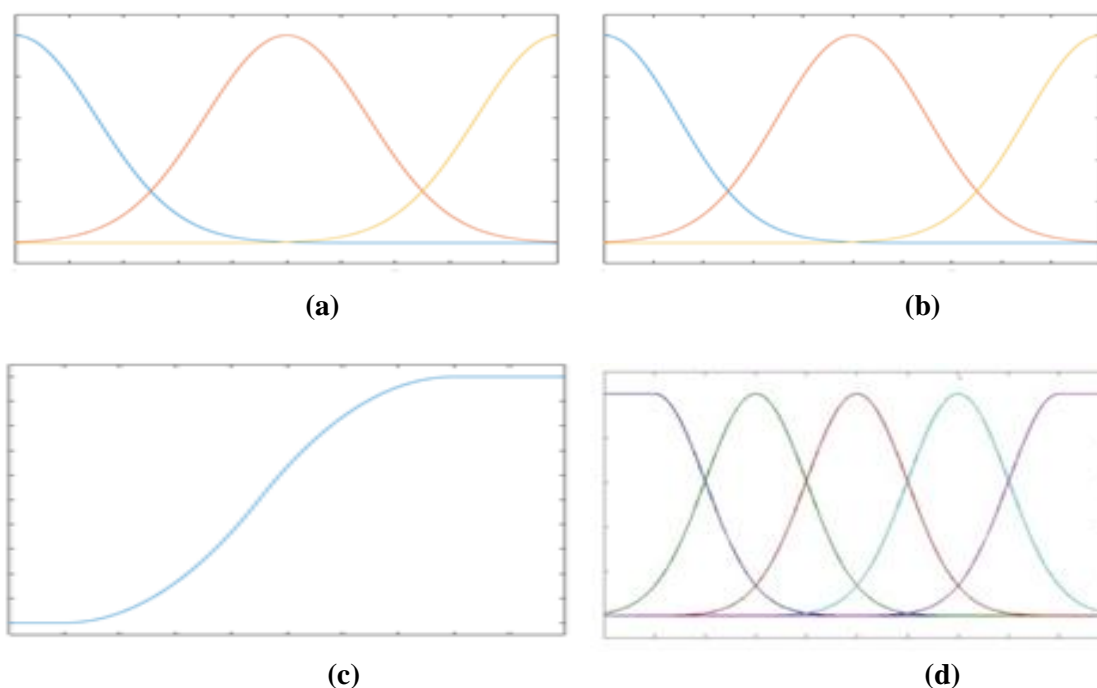


Figure II-9 : (a) Mesure de similarité SURF,(b) Mesure de similarité FREAK,
(c) L'appartenance à la région, (d) Mesure de similarité basée sur la région.

Voici les règles floues utilisées pour la proposition système pour déterminer la mesure de similarité basée sur la région. La méthode de défuzzification utilisée pour la sortie est la méthode centroïde.

1. Si (la mesure SURF est faible) et (la mesure FREAK est faible) alors (la mesure de similarité est très basse)
2. Si (la mesure SURF est moyenne) (la mesure FREAK est moyenne) et (r1 appartient) et (r2 appartient) alors (la mesure de similarité est moyenne)
3. Si (la mesure SURF est élevée) et (la mesure FREAK est élevée) alors (la mesure de similarité est très haut)
4. Si (la mesure SURF est moyenne) et (la mesure FREAK est moyenne) et (r1 n'appartient pas) et (r2 n'appartient pas) alors (la mesure de similarité est basse)
5. Si (la mesure SURF est moyenne) et (la mesure FREAK est moyenne) et (r1 n'appartient pas) et (r2 appartient) alors (la mesure de similarité est bas)
6. Si (la mesure SURF est moyenne) et (la mesure FREAK est moyenne) et (r1 appartient) et (r2 n'appartient pas) alors (la mesure de similarité est bas)

Pour chaque point d'intérêt correspondant de l'image de test à l'aide de SURF, FREAK, une mesure de similarité basée sur la région est obtenue en utilisant la logique floue discutée dans la section précédente. Les points d'intérêts sont dits être appariés sémantiquement s'ils se situent au-dessus d'un seuil et inégalé autrement. La mesure de similitude finale (RSM) est obtenue en utilisant l'équation suivante.

$$RSM = \frac{k}{n} \times 0.25 + \frac{1}{k} \sum_{i=1}^k FSM(k_1) \times 0.75$$

Où k est le nombre de points clés correspondants; n est le total nombre de points d'intérêts dans l'image de test et FSM est la similitude mesure obtenue à partir de la logique floue.

4. Conclusion

Nous avons pu constater dans ce chapitre les deux différents algorithmes et les étapes pour déterminer la mesure de similitude entre deux images qui peuvent réaliser une meilleure correspondance des fonctionnalités. L'objectif de ce travail est de fournir une description invariante aux changements pouvant survenir entre deux images, à la fin de ce chapitre nous avons présenté un algorithme flou pour l'association des points d'intérêt afin d'effectuer un bon résultat.

Le chapitre suivant présente la partie de la réalisation et de l'analyse. Il décrit le logiciel utilisé pour effectuer le développement, ainsi que les résultats des tests de notre étude.



CHAPITRE III :
RESULTATS ET
DISCUSSIONS

III. Résultats de Simulation

1. Introduction

Notre technique proposée est un algorithme basé sur la logique floue qui permet d'associer les points d'intérêts détectés entre deux images chevauchées, dans notre projet on s'intéresse sur un détecteur classique et simple pour la détection de points clés, combiner avec deux méthodes pour extraire les descripteurs autour les points détectés. Ce chapitre est dédié à la conception et le développement d'un système de vision par ordinateur pour associer les correctes correspondances entre les images. Dans ce qui suit, nous détaillerons les différentes étapes de la conception et de la réalisation de notre système, ainsi que les différents résultats obtenus. Pour évaluer les performances d'algorithme proposé, de nombreuses expériences ont été menées sur différentes paires d'images. L'organigramme suivant illustre les principales étapes de l'algorithme proposé :

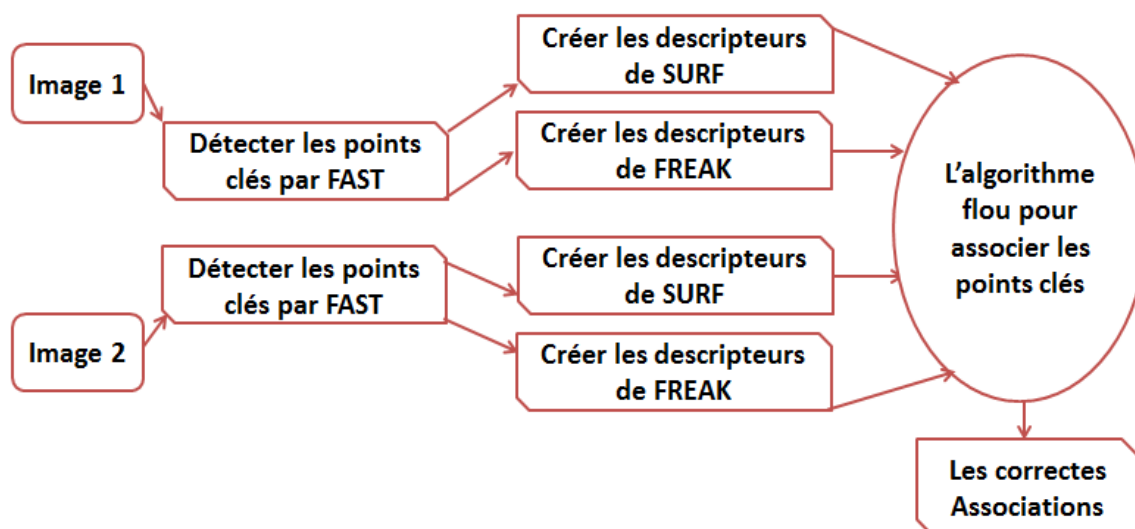


Figure III-1 : L'organigramme de notre technique proposé.

Selon la figure ci-dessus, après avoir acquis les images superposées; les caractéristiques sont détectées à l'aide de la technique FAST; qui est un simple détecteur de coin. Suivant; des descripteurs de SURF et FREAK doivent être créés pour toutes les primitives détectées dans les deux images. Enfin, l'algorithme flou proposé est appliqué pour donner un ensemble de correspondances correctes.

2. Environnement de développement

2.1. Environnement matériel

Nous avons utilisé un ordinateur qui à les caractéristiques suivants :

Type : PC/ DELL (Inspiron 15/ 3000 series).

Processeur : Intel® Core™i5-6300 CPU @ 2.40GHz 2.50GHz.

Mémoire installée (RAM) : 4.00Go.

Type Système : Système d'exploitation 64bits, processeur x64.

2.2. Environnement logiciel

MATLAB (« *matrix laboratory* ») est un langage de script émulé par un environnement de développement du même nom ; il est utilisé à des fins de calcul numérique. Développé par la société The MathWorks, MATLAB permet de manipuler des matrices, d'afficher des courbes et des données, de mettre en œuvre des algorithmes, de créer des interfaces utilisateurs, et peut s'interfacer avec d'autres langages comme le C, C++, Java, et Fortran.

Les utilisateurs de MATLAB (environ 4 millions en 2019) sont de milieux très différents comme l'ingénierie, les sciences et l'économie dans un contexte aussi bien industriel que pour la recherche. Matlab peut s'utiliser seul ou bien avec des *toolboxes* (« boîte à outils »).

Comme les images numériques ne sont que des matrices de certaines dimensions, toute opération peut être effectuée à l'aide de l'outil Matlab. Nous avons utilisé la dernière version de Matlab (Matlab2020.a), cette version regorge de fonctions de traitement d'image qui facilitent la mise en œuvre de notre algorithme.

3. Implémentation de Notre Algorithme Flou

3.1. Base de données

Plusieurs bases de données peuvent être utilisées pour tester notre algorithme, et la condition principale est d'avoir une région de chevauchement entre toutes les deux images successives. Les tailles des images peuvent également être réduites en utilisant la fonction *imresize* dans Matlab.

L'étape de départ de notre algorithme est la détection des points clés, pour plus de simplicité, nous avons proposé d'utiliser le détecteur de FAST, qui est détaillé dans la section suivante.

3.2. Détecteur de FAST

(FAST) [37], est un algorithme de détection de point conçu pour les applications temps réel. L'algorithme opère en deux étapes : dans la première étape, un test de segment basé sur les luminosités relatives est appliqué à chaque pixel de l'image traitée. Un point est considéré comme coin (point d'intérêt) lorsqu'un nombre N de points voisins situés sur un cercle sont plus clairs ou plus foncés que le point central. Les auteurs proposent d'utiliser un cercle discret de 16 pixels et de rayon 3 pixels (cf. Fig. 2.7). Le nombre N est posé à 12. Pour optimiser le temps de calcul les tests sont hiérarchisés de façon que les pixels voisins sont considérés dans l'ordre nord, sud, est, ouest (respectivement pixel 1; 9; 5; 13 dans la figure 2.7) dont trois au moins doivent être clairs pour que le point soit considéré comme un coin.

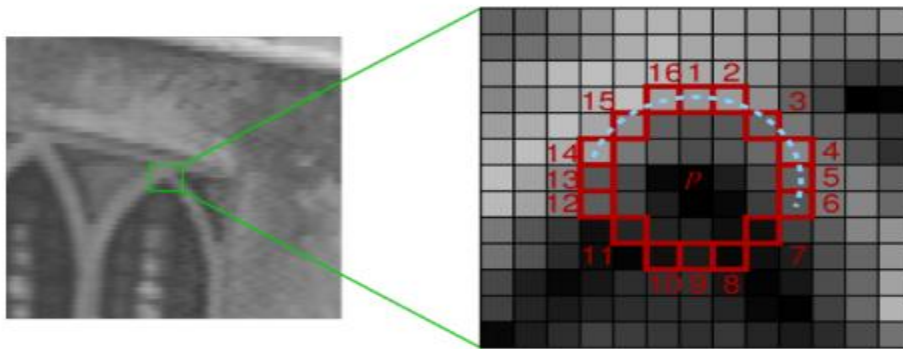


Figure III-2 : Test de segment appliqué sur un rayon de 3 pixels autour du pixel p .

Si ce test n'est pas satisfait, les pixels voisins restants ne sont pas vérifiés. La deuxième étape permet d'affiner et de limiter les multiples réponses du détecteur pour un même coin en supprimant les valeurs non maximales du score V à partir d'un seuil t :

$$V(p) = \max(\sum_{p1 \in E_{clair}} |I(p) - I(p1)| - t, \sum_{p1 \in E_{foncé}} |I(p) - I(p1)| - t) \quad (\text{III-1})$$

Où $(E_{clair}; E_{foncé})$ correspondent respectivement à l'ensemble des points plus claires (plus foncé) que le point candidat. Et $I(p1)$ correspond à l'intensité du pixel candidat. Une version améliorée du détecteur FAST proposé par le même auteur remplace les tests statistiques par un arbre de décision construit à partir d'un apprentissage sur un ensemble d'image. Ce détecteur offre de bons résultats pour un temps de calcul très rapide. Néanmoins le détecteur FAST est très sensible au flou et au changement d'échelle. L'extension à l'invariance à l'échelle est assurée par le détecteur BRISK [38], qui combine l'espace échelle et l'approche FAST.

3.3. Association de Points d'Intérêts par l'Algorithme Floue

Après la détection des points d'intérêt dans chaque image, les descripteurs discutés dans le chapitre précédent sont créés pour tous les points détectés. Pour les descripteurs SURF, nous devons appliquer la distance euclidienne entre les descripteurs de l'image 1 et de l'image 2 et pour les descripteurs FREAK; nous devons appliquer la distance Hamming.

- **Distances Géométriques D'association**

Considérons deux vecteurs X et Y :

$$X = (x_1, x_2, x_3, x_4 \dots x_n) \text{ et } Y = (y_1, y_2, y_3, y_4 \dots y_n)$$

- i. **Distance Euclidien**

Pour l'application de distance euclidien $Dist_{Euc}$; les vecteurs X et Y doivent être décimales et la formule mathématique est donnée par l'équation suivante :

$$Dist_{Euclidien}(X, Y) = \sqrt{(x_1 - y_1)^2 + \dots + (x_n - y_n)^2} \quad (III-2)$$

- ii. **Distance Hamming**

La distance Hamming est une opération logique XOR (\oplus) entre deux vecteurs binaires; donc les descripteurs X et Y doivent être binaires et la formule mathématique est donnée par l'équation suivante :

$$Dist_{Ham} = X \oplus Y \quad (III-3)$$

Selon les distances calculée entre les descripteurs des deux images, les c correctes associations des points clés peuvent être trouvées comme indiqué par le diagramme suivant :

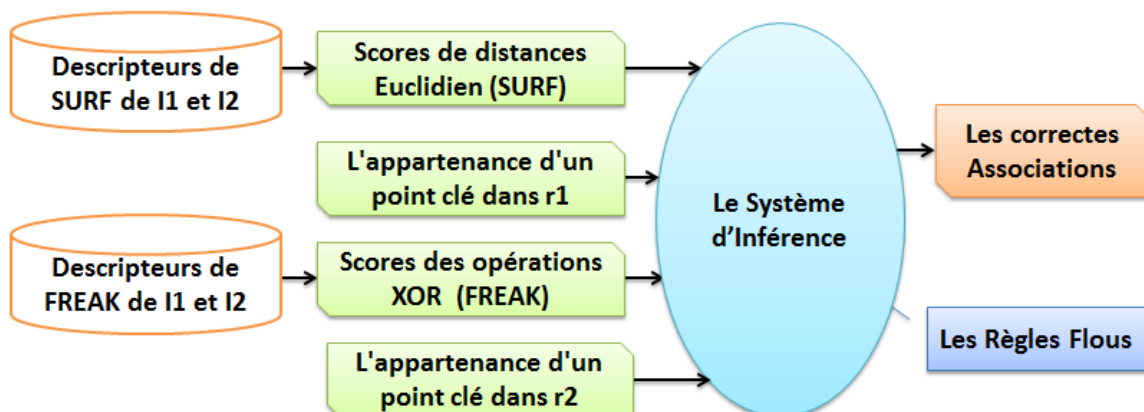


Figure III-3 : Les étapes de notre algorithme pour trouver les associations correctes.

3.4. Les résultats Obtenus

Les figures suivantes montrent les images superposées utilisées pour nos tests, avec les résultats de la détection de points d'intérêts à l'aide de l'algorithme FAST et les résultats de la association des points d'intérêts à l'aide de l'algorithme flou :



Figure III-4 : Les images chevauchées utilisées de la première scène.



Figure III-5 : Points d'intérêt détectés de la première scène à l'aide du détecteur FAST.

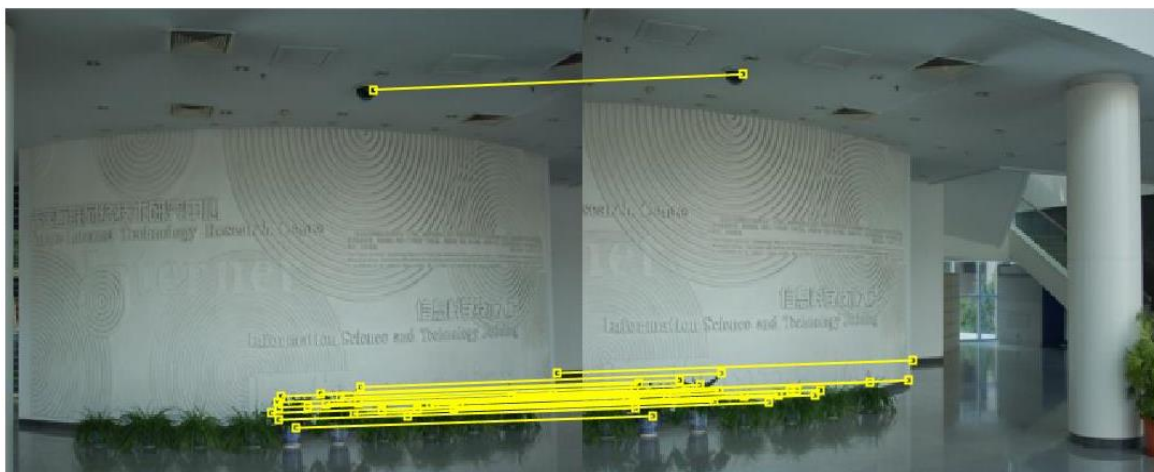


Figure III-6 : Points d'intérêts associés entre les images par l'algorithme flou de 1^{ère} scène.



Figure III-7 : Les images chevauchées utilisées de la deuxième scène.

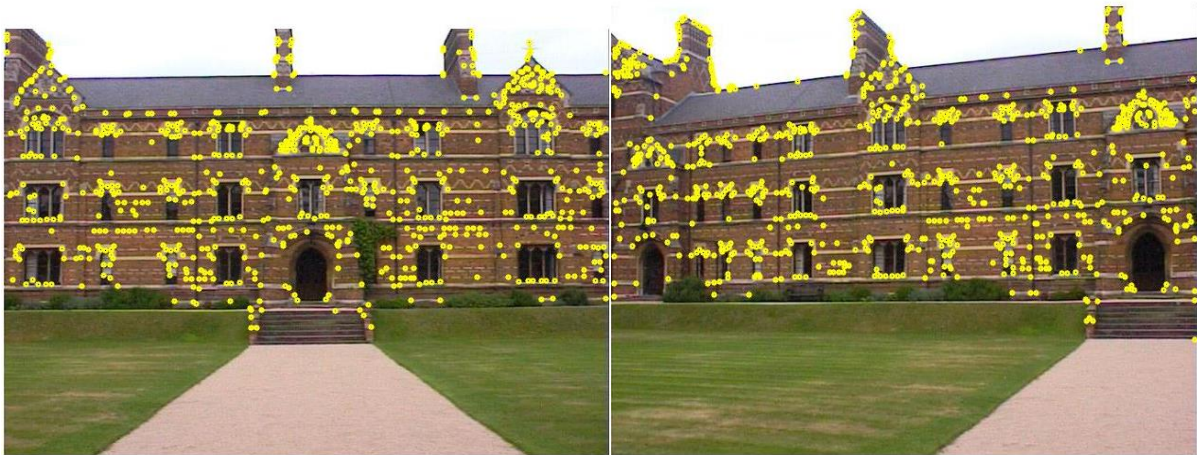


Figure III-8 : Points d'intérêt détectés de la deuxième scène par détecteur FAST.

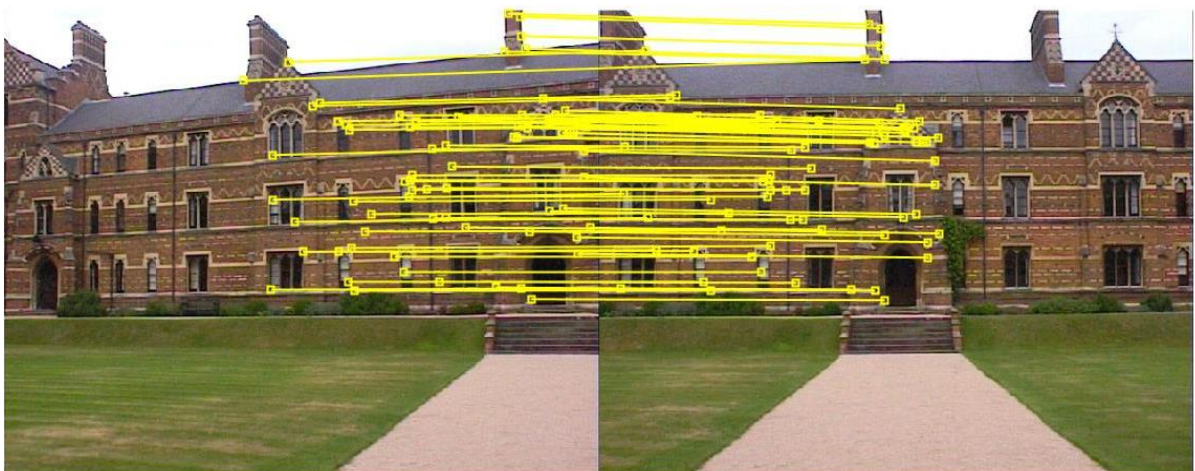


Figure III-9 : Points d'intérêts associés entre les images par l'algorithme flou de 2^{ème} scène.



Figure III-10 : Les images chevauchées utilisées de la troisième scène.

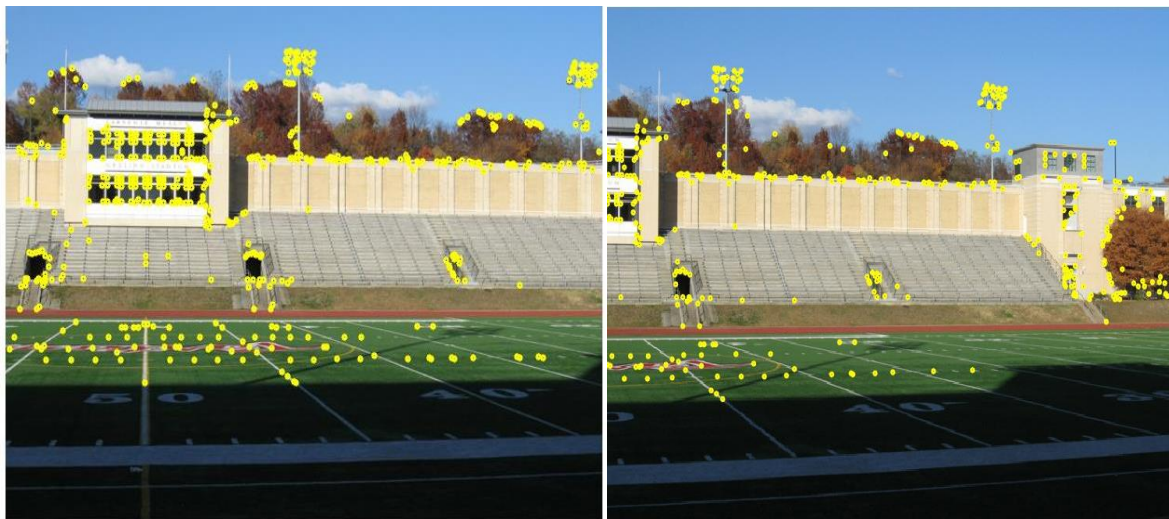


Figure III-11 : Points d'intérêt détectés de la troisième scène à l'aide du détecteur FAST.

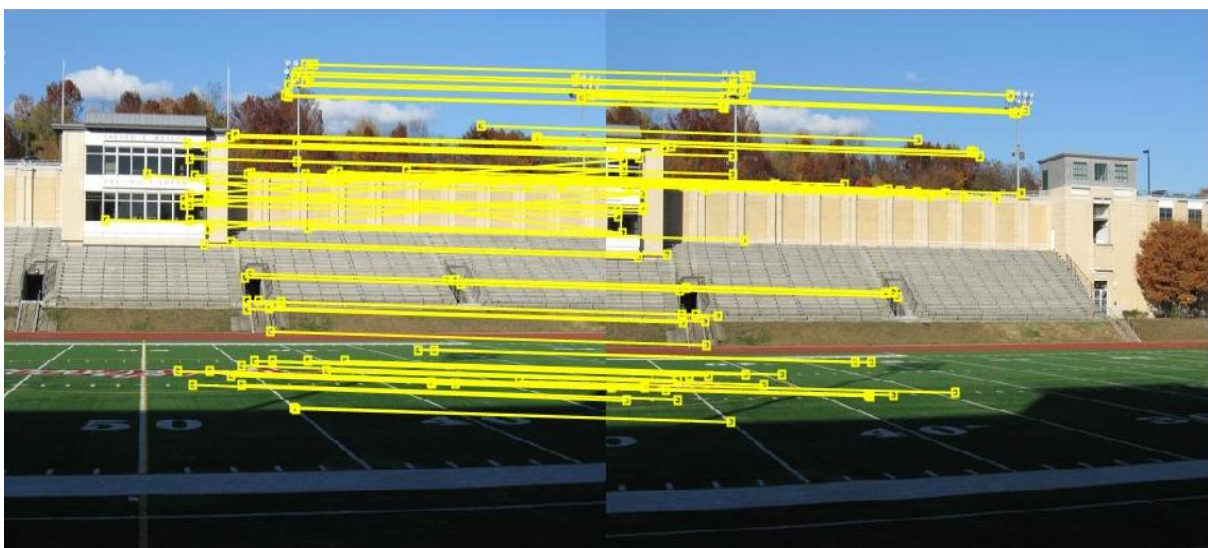


Figure III-12 : Points d'intérêts associés entre les images par l'algorithme flou de 3^{ème} scène.



Figure III-13 : Les images chevauchées utilisées de la quatrième scène.



Figure III-14 : Points d'intérêt détectés de la quatrième scène à l'aide du détecteur FAST.



Figure III-15 : Points d'intérêts associés entre les images par l'algorithme flou de 4^{ème} scène.

3.5. Discussion des Résultats

En utilisant l'image de gauche de la quatrième scène, le tableau suivant illustre l'effet de la valeur de seuil FAST sur le nombre de primitives détectées et le temps de traitement. Dans le tableau III-1; nous illustrons la dépendance entre le seuil et le nombre de points clés et le temps d'exécution. A titre d'illustration, on fait varier le seuil dans la fourchette seuil $\in [6; 100]$; le seuil haut est fréquemment utilisé afin de garantir un nombre minimum de détections avec un temps de traitement court.

Tableau III-1:L'effet du seuil FAST sur les primitives détectées.

Seuil	Points Clés	Temps (s)
100	7	0.4662
75	116	0.3270
50	1356	0.6143
25	8883	0.8797
12	20335	0.9219
6	28642	0.9183

Pour les régions contrastées; certaines scènes particulièrement nettes peuvent présenter un grand nombre de détections même pour des seuils aussi élevés que seuil = 50. Dans notre algorithme; en raison des grandes régions de chevauchement entre les images utilisées et de la bonne répétabilité du détecteur FAST, une valeur de seuil de 75 sera efficace.

- **La répétabilité**

La répétabilité représente la capacité de localiser le même point clé dans la région de chevauchement entre deux images. Ce paramètre peut être affecté par les changements d'éclairage et le bruit d'imagerie [39]. La valeur de répétabilité peut être calculée par l'équation III-2:

$$\text{répétabilité} = \frac{\text{Primitives répétées}}{\text{Primitives dans la région de chevauchement}} \quad (\text{III-2})$$

La valeur de répétabilité plus élevée représente les meilleures performances du détecteur de points clés.

Tableau III-2: Calcul de répétabilité entre les images chevauchées.

Les scenes	I 1	I 2	Répétabilité
Scene # 1	40	57	0.059
Scene # 2	808	784	0.061
Scene # 3	508	406	0.071
Scene # 4	942	799	0.073

La plupart des détecteurs d'angle se caractérisent par une répétabilité élevée; ce qui est le cas avec le détecteur FAST comme indiqué dans le tableau ci-dessus. Ce paramètre change en fonction du rapport de chevauchement et de la texture de la scène.

- **Pourcentage De Correspondance Correcte « Recall »**

Ce critère d'évaluation [40] représente la relation entre le nombre de correspondances correctes et le nombre de fausses correspondances obtenues pour la paire d'images. Ce critère peut être donné par l'équation III-3:

$$Recall = \frac{\text{Les association correctes}}{\text{Toutes Les associations}} \quad (\text{III-3})$$

La valeur de *recall* idéale est 1; dans ce cas, toutes les primitives correspondantes sont des correctes; et aucune existence de fausses correspondances.

Tableau III-3: Calcul de pourcentage de correspondance correcte.

Les scenes	I 1	I 2	Recall
Scene # 1	40	57	0.62
Scene # 2	808	784	0.68
Scene # 3	508	406	0.71
Scene # 4	942	799	0.77

Dans les quatre scènes, une valeur élevée de rappel a été obtenue; ce qui signifie que notre algorithme fournit un bon nombre d'associations correctes.

4. Conclusion

Dans ce chapitre, nous avons appliqué notre algorithme proposé de correspondance floue sur différentes images, et nous avons constaté qu'il donnait des résultats acceptables en termes de certains critères de performance.



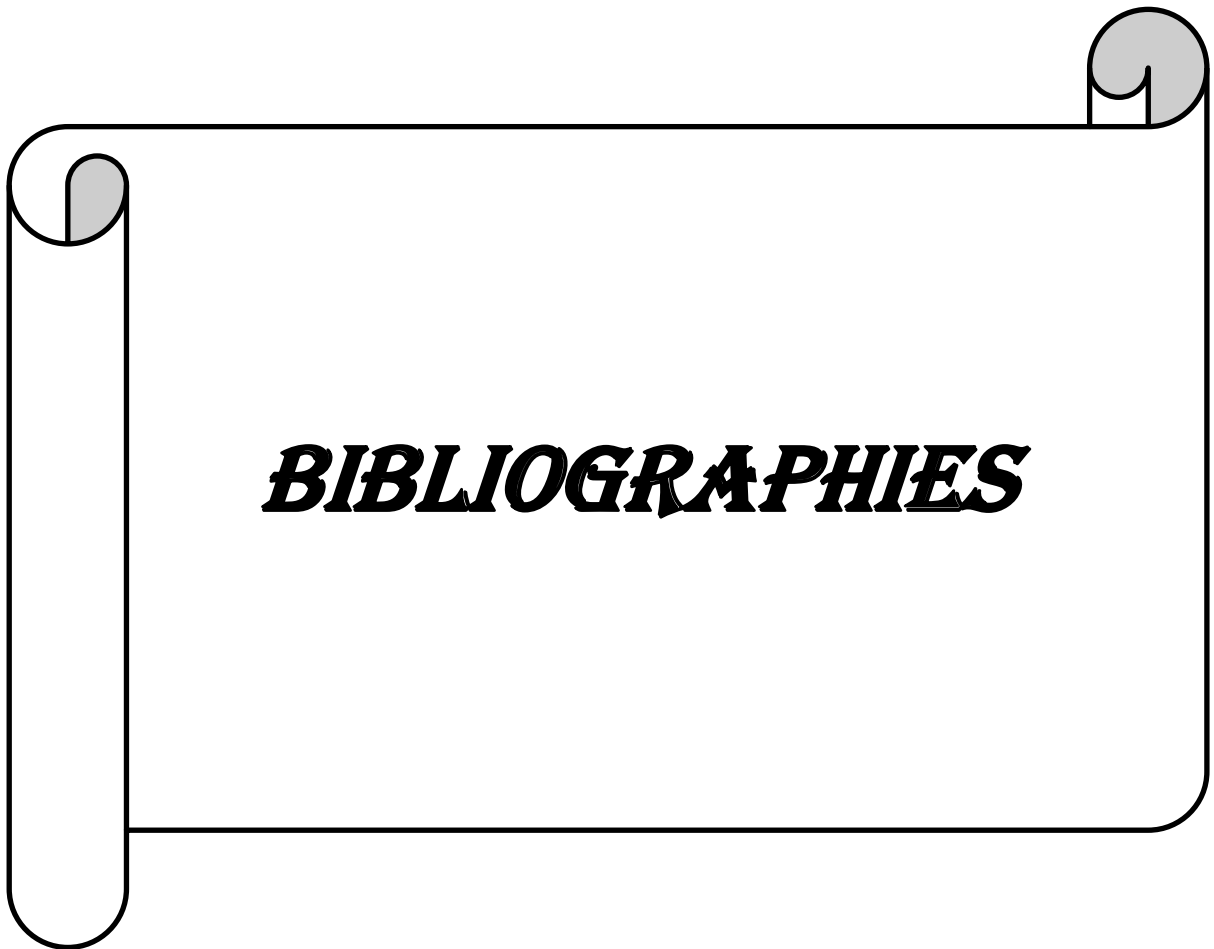
CONCLUSION
GENERALE

Conclusion Générale

Le processus de correspondance des caractéristiques de l'image joue un rôle important dans le nombre d'applications modernes liées au traitement des images numériques dans les domaines scientifiques modernes. Les systèmes flous et la théorie des ensembles flous, fournissent un complément riche et significatif à la logique standard. Les mathématiques générées par ces théories sont cohérentes peuvent être une généralisation de la logique classique. Les applications qui peuvent être générées ou adaptées à la logique floue sont très variées et offrent l'opportunité de modéliser des conditions qui sont intrinsèquement définies de manière imprécise, malgré les préoccupations des logiciens classiques.

La principale contribution de notre travail est l'implication de la théorie de la logique floue dans l'appariement des caractéristiques de l'image. Nous avons suggéré l'utilisation des descripteurs SURF et FREAK pour construire les variables d'entrée floues. La sortie de notre algorithme est soit une correspondance correcte, soit une correspondance fautive. Dans chaque scène de teste, deux images ont été acquises et primitives ont été détectés et associés différence entre les deux images. Les résultats obtenus montrent l'efficacité de l'algorithme proposé, dans lequel seules les correspondances correctes sont fournies et les fausses sont éliminées.

La correspondance des caractéristiques est une étape clé dans les applications de traitement d'image importantes telles que la classification d'images, l'enregistrement d'images, le mosaïquage d'images et la reconstruction 3D, nous vous recommandons donc d'utiliser nos algorithmes dans ces applications; nous proposons également d'utiliser d'autres variables d'entrée floues et un autre système d'inférence. Enfin, nous vous recommandons d'utiliser d'autres types d'images telles que les images thermiques.



BIBLIOGRAPHIES

Références

- [1] Mohamed salah BEKKOUCHE, Ahmed ABOUBOU « La criminalistique d'appareils photo : « La mise en place d'un système de preuve en forensic » » Mémoire de master universite kasdimerbah ouargla, 2017
- [2] .Sandeli, traitement d'images par des approches bio-inspirées application à la segmentation d'images, université constantine 2. 2014.
- [3] F. Schaffalitzky, A. Zisserman, Multi-view Matching for Unordered Image Sets. In Proc. ECCV. (2002), pp. 414–431. 2. T. Tuytelaars, L. Van Gool, Matching widely separated views based on affine invariant regions. Int J Comput Vis 1(59), 61–85 (2004).
- [4] C.Houassine, segmentation d'images par une approche biomimétique hybride. ccccc université universite m'hamedbougara- boumerdes. 2012.
- [5] Henri Maître, « 10. La détection de contour dans les images », dans Henri Maître (dir.), Le traitement des images, Hermès Sciences, 2003.
- [6] elisearnaud « Extractions de caractéristiques - les points d'intérêts ».
- [7] www.pantechsolutions.net.(s.d). consulte le 18 08,2020, sur pantechsolutions; <https://www.pantechsolutions.net/blog/matlab-code-for-region-of-interest-in-image/>
- [8] KrystianMikolajczyk, TinneTuytelaars, CordeliaSchmid, Andrew Zisserman, J. Matas, F. Schaffalitzky, T. Kadir et L. Van Gool, « A comparison of affine region detectors », International Journal of Computer Vision,
- [9] Da Rugna, J. etKonik, H. "Color interest points detector for visual information retrieval". LIGIV, Université Jean Monnet, St Etienne, France, 2002.
- [10] SOLTANI DJAHIDA « point interet pour la detection des logos des vehicules » mémoire master informatique, universite de larbitebessi-tebessa.
- [11] T. Lindeberg "Scale selection properties of generalized scale-space interest point detectors", Journal of Mathematical Imaging and Vision, Volume 46, Issue 2, pages 177-210, 2013.
- [12] T. Lindeberg "Image matching using generalized scale-space interest points", Journal of Mathematical Imaging and Vision, volume 52, number 1, pages 3-36, 2015.
- [13] Schmid, Cordelia; Mohr, Roger; Bauckhage, Christian (1 January 2000). "Evaluation of Interest Point Detectors".

- [14] Minoru, M., Kunio, K.: Fast Template Matching Based on Normalized Cross Correlation Using Adaptive Block Partitioning and Initial Threshold Estimation. In: IEEE International Symposium on Multimedia, pp. 196–203. IEEE Press. 2010.
- [15] Sigmis: A Feature Selection Algorithm Using Correlation Based Method E. Chandra Blessie¹, and E. Karthikeyan².
- [16] Alhwarin, Faraj. Fast and robust image feature matching methods for computer vision applications. University of Bremen, 2011.
- [17] Lecture 4.2 Feature matching Trym Vegard Haavardsholm
- [18] V. Rehrmann. Object oriented motion estimation in color image sequences. In Proceedings European Conference on Computer Vision, volume 1, Freiburg, Germany.
- [19] J. Shiand et C. Tomasi, “ Good features to track ” .In Proceeding of IEEE Conference on Computer Vision an Pattern Recognition, Seattle. Pages 593–600. 1994
- [20] GodjevacJ.,Idées nettes sur la logique floue, PPUR : Lausanne, collection informatique, 1999, 128p.
- [21] Franck Dernoncourt, La logique floue entre raisonnement humain et intelligence artificielle, mémoire de master, Janvier 2011
- [22] z gharbifatimazahra -ben namiahalima « La logique floue pour Classification Des Feuilles de vigne » mémoire master,université de ouargla 2016
- [23] Université Mouloud MAMMERI deTizi-Ouzou mémoire magisterpour Traitement d’Images et Reconnaissance des Formes Présenté par : MelleSBILI Lila.
- [24] fuzzy_logic_dvp_2011.pdf
- [25] Krim Mohamed Housseem Eddine « Etude de l’algorithme FCM pour la segmentation des images satellite landsat 5 » mémoire master Université Dr. Tahar Moulay -SAIDA-]
- [26] MESLEM, mémoire de magister « Développement d’une approche d’intégration de la logique floue dans la représentation des connaissances », Institut National de formation en Informatique (I.N.I) Oued-Smar, Alger, 08-mars-2009.
- [27] MEZAACHE NOURDDINE, Mémoire magister « Commande d’un bras manipulateur par logique floue et les algorithmes génétiques » ,université de setif
- [28] P. M. Panchal, S. R. Panchal, S. K. Shah, "A Comparison of SIFT and SURF ", International Journal of Innovative Research in Computer and Communication Engineering Vol. 1, Issue 2, April 2013
- [29] Mikolajczyk, K., Schmid, C.: Indexingbased on scale invariant interest points. In: ICCV. Volume 1, (2001).

- [30] P M Panchal, S R Panchal , S K Shah (13 April), A Comparison of SIFT and SURF, International Journal of Innovative Research in Computer and Communication Engineering
- [31] H. Bay, T. Andreasess, Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. Computer Vision and image understanding, vol. 110, n°3, 2008. p346-359
- [32] Herbert Bay , Andreas Ess , Tinne Tuytelaars , and Luc Van Gool . Speeded-Up Robust Features (SURF). ETH Zurich, BIWI Sternwartstrasse 7 CH-8092 Zurich Switzerland bK.U. Leuven, ESAT-PSI Kasteelpark Arenberg 10 B-3001 Leuven Belgium
- [33] E. Tola, V. Lepetit, and P. Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 32(5):815–830, 2010. 3
- [34] Franck Deroncourt, La logique floue entre raisonnement humain et intelligence artificielle, mémoire de master, Janvier 2011.
- [35] R. Ortiz, “FREAK: Fast Retina Keypoint,” in Proceeding of CVPR. Washington, DC, USA: IEEE Computer Society, , pp: 510–517. 2012.
- [36] Yao, Q., & Xu, X.. Freak Descriptor With Spatial Pyramid Kernel For Scene Categorization. Inte. Conference on Mechatronics, Electronic, Industrial and Control Engineering Published by Atlantis Press. (2015).
- [37] TRAJKOVIĆ, Miroslav et HEDLEY, Mark. Fast corner detection. Image and vision computing, 1998, vol. 16, no 2, p. 75-87.
- [38] Leutenegger, S, Margarita C, and Roland Y. S. "BRISK: Binary robust invariant scalable keypoints." 2011 International conference on computer vision. Ieee, 2011.
- [39] Ehsan, Shoaib, et al. "Improved repeatability measures for evaluating performance of feature detectors." Electronics letters 46.14 (2010): 998-1000.
- [40] Hassaballah, M., Abdelmgeid, AA., and Hammam, A. 'Image Features Detection, Description and Matching'. Springer International Publishing Switzerland. 2016.

Résumé

Les points clés d'une image font référence aux pixels intéressants avec des informations significatives pour les tâches de traitement d'image. La mise en correspondance de primitives est une étape nécessaire dans la plupart des applications de vision par ordinateur, et divers algorithmes ont été développés par les chercheurs à cette fin. La plupart des algorithmes proposés sont basés sur la comparaison et l'analyse des correspondances de points entre des images qui se chevauchent en utilisant certaines mesures. Trouver de bonnes correspondances des paires de points à partir de deux ensembles de points donnés est l'un des principaux problèmes de la correspondance des points caractéristiques. Par conséquent, dans ce travail; nous avons proposé d'utiliser la technique d'association floue basée sur les descripteurs SURF et FREAK et nous avons testé nos approches sur différentes images. Les résultats obtenus ont été efficaces en termes de mesures de performance.

Mots clés: Mise en correspondance, Logique floue, SURF , FREAK

Abstract

Key points in an image refer to the interesting pixels with meaningful information for image processing tasks. Feature matching is necessary step in most computer vision applications, and various algorithms have been developed by researchers for this purpose. Most of the proposed algorithm are based on comparing and analyzing point correspondences between overlapping images by using some measures. Finding good matches of the pairs of points from two given point sets is one of the main issue of feature point matching. Therefore, in this work ; we have proposed to utilize fuzzy matching technique based on SURF and FREAK descriptors and we have tested our approaches on different images. The obtained results were efficient in terms of performance measures.

Key words: Features matching, Fuzzy logic, SURF , FREAK.

ملخص

النقاط المفتاحية في الصورة هي بيكسلات مهمة تحتوي على معلومات مفيدة من أجل تطبيقات معالجة الصور. تعد مطابقة النقاط الدالة خطوة ضرورية في معظم تطبيقات رؤية الكمبيوتر ، وقد طور الباحثون خوارزميات مختلفة لهذا الغرض. تعتمد معظم الخوارزميات المقترحة على مقارنة وتحليل العلاقات النقطية بين الصور المتداخلة باستخدام بعض المقاييس. يعد العثور على التطابقات الجيدة لأزواج النقاط من مجموعتين محددتين من النقاط أحد المشكلات الرئيسية لمطابقة النقاط الدالة، لذلك ، في هذا العمل ؛ لقد اقترحنا استخدام تقنية المطابقة الضبابية بناءً على واصفات SURF و FREAK وقمنا باختبار مناهجنا على صور مختلفة. كانت النتائج المتحصل عليها فعالة من حيث مقاييس الأداء.

مفتاحية كلمات : مطابقة النقاط الدالة ، المنطق الضبابي ، SURF , FREAK .