

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad Dahlab Blida



Faculté des sciences

Département d'informatique

Mémoire Présenté par :

BOUYAKOUB Hafidha ERROUKRMA Asma

En vue d'obtenir le diplôme de master

Domaine : Mathématique et informatique

Filière : Informatique

Spécialité : Informatique

Option : Génie des System Informatique

Sujet :

Détection de copié vidéo basée sur le contenu Spatio-temporel

Soutenu le :

M.	Président
M.	Examineur
M.	Examineur
Mr Kameche Abdelah Hicham	Promoteur
Mme Ait Sadi Karima	Encadreur

Promotion
2015/ 2016



Remerciement

Tout d'abord, nous souhaitons manifester mes sincères remerciements à ALLAH le tout puissant qui nous a donné la force, la patience, le courage, la volonté et la santé durant toutes ces années d'étude pour aller jusqu'un bout de ce travail.

Nous tenons avant tout à remercier Mme K. Ait Sadi, Mme I. Bouchair et Mr A. Kamèche qui nous a permis de poursuivre notre projet, leur présence et leur aide et surtout pour ses conseils utiles et sa modestie.

Nous remercions aussi chaleureusement le membre de jury, pour l'honneur qui ils nous ont fait en acceptant d'évaluer notre travail.

Au terme de ce travail, nous voudrions adresser nos vifs remerciements à tous nos enseignants de département d'informatique.

Nous ne pouvons clore cette partie sans remercier nos familles, nos amis, et plus particulièrement nos parents pour nous avoir supportés et soutenus avec beaucoup de patience et de tendresse.

Merci à tous encore.

ERROUKMA

ASMA

BOUYAKOUB

HAFIDHA





Dédicace

Je dédie ce travail à mes très chers parents, pour leur soutien et tous les efforts qu'on m'a donnée le long de mon parcours et je leurs souhaite bonne santé et longue vie.

A toute la famille BOUYAKOUB.

A me chère sœur MERIEME.

A tous mes frères.

A tous mes proches.

A tous mes chères amies sans oublier mon binôme ASMA.

A tous ceux que j'aime et qui m'aiment.

Ce mémoire leur est dédié.

BOUYAKOUB

НАФИДНА



Dédicace

Je dédie ce travail à mes très chers parents, pour leur soutien et tous les efforts qu'on m'a donnée le long de mon parcours et je leurs souhaite bonne santé et longue vie.

A toute la famille ERROUKRMA et la famille ELLAHOUEL.

A mes chères sœurs HANANE ET CHAIMA.

A tous mes cousins et cousines.

A tous mes oncles et mes tantes.

A mes grands parents et grandes mères.

A tous mes proches.

A tous mes chères amies sans oublier mon binôme HAFIDHA

A tous ceux que j'aime et qui m'aiment.

Ce mémoire leur est dédié.

❖ **BOUYAKOUB HAFIDHA**

❖ **ERROUKRMA ASMA**

ERROUKRMA

ASMA

Résumé

De nos jours, Les images numériques peuvent être copiées, modifiées et réutilisées avec facilité. La détection de copies d'image a donc émergé comme un domaine de recherche important et prometteur dans la gestion des droits numériques. Dans ce mémoire, nous avons étudié les systèmes de détection de copie d'images basées sur le contenu (CBCD). Il repose sur l'extraction des caractéristiques compactes des images de la séquence vidéo. La détection utilise des mesures de similarité pour la décision de la copie ou non copie de la vidéo.

Plus précisément, nous avons présenté dans ce travail les résultats des tests effectués par l'application des descripteurs locaux LBP et ses variantes. Nous avons par la suite démontré que notre système CBCD basé sur le descripteur LBP-TOP uniforme avec un voisinage de 8 et amélioré par l'introduction d'un prétraitement log polaire est robuste après l'avoir testé sur la base de données TRECVID 2009. L'efficacité et la robustesse de la méthode développée sont montrées pour diverses attaques comme la compression H.264/AVC, l'ajout de bruit, la rotation et le retournement sont démontrées au travers des expérimentations réalisées.

Mots clés

Détection de copie vidéo, Descripteur spatio-temporelle, Descripteur LBP, Similarité, Attaques.

Abstract

Nowadays, digital images can be copied, modified and reused with ease. The detection of image copies has emerged as an area of important and promising research in digital rights management. In this paper, we studied the image copy detection systems based on the content (CBCD). It is based on the extraction of signatures compact images of the video sequence. Detection uses similarity measures for the decision to copy or not copy the video.

Specifically, we presented in this work the results of tests performed by the application of local descriptors LBP and its variants. We subsequently demonstrated that our CBCD based descriptor uniform LBP-TOP system with a neighborhood of 8 and improved by the introduction of a polar log preprocessing is robust after testing on TRECVID 2009 database. the effectiveness and robustness of the developed method are shown for various attacks such as compression H.264 / AVC, adding noise, rotation and flipping are demonstrated through experiments performed.

Keywords

Copy Detection Video, Spatio-Temporel Descriptor, LBP Descripteur, Similarite, Attacks.

المخلص

في الوقت الحاضر، الصور الرقمية يمكن نسخها أو تعديلها وإعادة استخدامها بكل سهولة. وقد برز الكشف عن نسخ صورة كمجال للبحث مهم وواعد في إدارة الحقوق الرقمية. في هذه المذكرة، درسنا أنظمة الكشف عن نسخ من الصورة التي يقوم على استخراج ميزات الصور المدمجة في تسلسل الفيديو. يستخدم الكشف تدابير التشابه، على أساس المحتوى قمنا بدراسة أنظمة كشف النسخ.

على وجه التحديد، قدمنا في هذا العمل بعرض نتائج الاختبارات التي يؤديها الواصف المحلي LBP ومشتقاته .

بينما أن نتائج الواصف المحلي LBP-TOP أظهرت فعالية ومتانة. وحسناها مع الدوران باستعمال

polaire. أثبتنا بعد ذلك أن طريقتنا المقترحة فعالة مع التغيرات المختلفة مثل ضغط، الضوضاء، تغيير اللون... الخ

Sommaire

Introduction général	1
Chapitre 1 : Généralité sur le système de détection de copie vidéo	4
Introduction	4
I. Définition d'une vidéo	5
I.1 Caractéristique d'une vidéo.....	6
I.1 .1 La couleur.....	6
I .1.2 La texture.....	8
I.1.3 La forme.....	10
I.1.4 Le mouvement.....	10
I.2 La résolution d'une image.....	10
I.3 Résolution temporelle d'une vidéo	10
I.4 Les différents formats de fichiers vidéo	11
I .5 Une copie vidéo	12
II. Système de détection de copie vidéo	12
II.1 Tatouage numérique (watermarking).....	12
II.2 Détection de Copie Basé sur le Contenu (CBCD).....	12
II.2.1 Les étapes de la détection de copie basée sur le contenu (CBCD).....	13
II.2.2 La similarité.....	14
II.2.3 Classifications des méthodes de CBCD.....	18
Conclusion	22
Chapitre 2 : Les descripteurs Locaux	24
Introduction	24
I. Le principe d'un descripteur local.	25
I.1 LPQ (Local Phase Quantization)	26
I.2 BSIF (Binarized Statistical Image Features).....	27
I.2.1 BSIF_TOP (Binarized Statistical Images Features Three Orthogonal Planes).....	29
I.2.2 MBSIF (muti-scale BSIF).....	30
I.3 HOG (Histogram of oriented gradients).....	31
I.4 SIFT (scale-invariant feature transform).....	32
I.5 DAISY.....	34

I.6 LBP (local binary pattern).....	35
I.6.1 Principe.....	35
I.6.2 LBP Uniforme (Uniform Patterns).....	37
I.6.3 LBP de rotation invariant.....	39
I.7 LBP Spatio-temporel.....	41
I.7.1 VLBP (Volume Locale Patterns Binaires).....	41
I.7.2 Volume Local Binary Patterns de rotation invariante.....	43
I.7.3 LBP-TOP (LBP- Tree Orthogonal Plan)	45
I.7.4 La différence entre les descripteurs VLBP et LBP-TOP.....	46
Conclusion.....	47
Chapitre 3 : Résultats et Evaluation.....	48
Introduction.....	48
I. Conception du système de détection de copie vidéo	48
I.1 Base de données de test.....	49
I.2 Le module de pré-traitement	51
I.3 Le module d'extraction de caractéristiques.....	51
I.4 Le module de comparaison	51
I.5 Le module de décision et performances du système réalisé	52
II. Résultats expérimentaux par l'application des descripteurs LBP et ses variantes dans le domaine spatial.....	53
II.1 Performances du système dans le domaine spatial	57
II.1.1 Les résultats du descripteur LBP de base	57
II.1.2 Les résultats du descripteur LBP uniforme.....	59
II.1.3 Les résultats du descripteur LBP rotationnel.....	62
II.1.4 Les résultats du descripteur LBP uniforme rotationnel.....	65
II.2 Performances du système dans le domaine spatio-temporel	68
II.2.1 Résultats obtenus par l'application du descripteur VLBP.....	69
II.2.2 Résultats obtenus par l'application du descripteur VLBP rotationnel.....	71
II.2.3 Résultats obtenus par l'application du descripteur LBP-TOP non uniforme.....	73
II.2.4 Résultats obtenus par l'application du descripteur LBP-TOP uniforme.....	75
II.2.5 Comparaison des résultats.....	80

III. La décision.....	80
III.1 Application des arbres de décision dans le processus de détection.....	81
III.2 Application Le seuil de détection.....	81
Conclusion.....	82
Chapitre 4 : Implémentation et Capteurs d'écran	84
Introduction.....	84
I. Langage et Outils Utilisée	84
I.1 le Matlab	84
I.2 Le langage Java.....	84
I.3 Eclipse.....	85
I.4 API (Application programming interface) utilisées.....	85
I.4.1 Matlabcontrol.....	85
I.4.2 Java Media Framework (JMF).....	85
II. la simulation et les captures d'écran	86
Conclusion.....	89
Conclusion général.....	90
Annexe.....	91

Introduction général

L'apparition et l'évolution rapide des nouvelles technologies numérique dans les appareils d'acquisition numériques et le multimédia (téléphones portables, Internet, la télévision numérique, etc) a ses avantages et ses inconvénients. D'une part cette évolution a conduit à une augmentation explosive des données multimédia disponibles dans le domaine public et exploités par de nombreuses applications telles que journalistiques, éducatives, médicales, vidéosurveillance et autres. D'autres part, elle a acheminé à la création et au partage de grandes quantités de données vidéo qui ont conduit à la création des copies multiples d'un même contenu vidéo. En conséquence, Il ya un besoin croissant pour les applications vidéo de mettre en place des outils permettant d'une part de trouver rapidement la vidéo recherchée dans de grandes bases de données, et d'autre part protéger le contenu en élaborant des mécanismes de protection du contenu. Parmi celles-ci, on trouve celles qui sont basées sur la détection de copie vidéo basé le contenu (CDVC),

Ce dernier porte sur la recherche des vidéos par le contenu pour la détection de copies vidéo dans les bases de données multimédias. Aujourd'hui, des logiciels de traitement des vidéos puissants sont disponibles au public pour permettre les manipulations ou les modifications des vidéos. Le succès de l'Internet et le dispositif de stockage numérique rentable permettent de reproduire, transmettre, et distribuer du contenu numérique de manière assez facile. Ainsi, la protection des droits de propriété intellectuelle est devenue une question juridique cruciale. Plus particulièrement, la détection des copies de médias numériques est une exigence de base pour la protection de la propriété intellectuelle.

Une des mécanismes de protection du contenu pour protéger la propriété intellectuelle est le tatouage numérique qu'on appelle aussi le watermarking. Ce dernier a été proposé pour la protection du droit d'auteur et les empreintes digitales. L'idée de base est d'insérer des informations dans le contenu vidéo. Certaines marque sont visibles (par exemple, le texte ou le logo du producteur ou diffuseur), tandis que d'autres sont cachés dans le contenu d'une façon que l'œil humain ne les perçoit pas. Aujourd'hui, tous les films DVD, les jeux vidéo, etc. ont des empreintes digitales qui prouvent la propriété du matériau. Comme inconvénient du tatouage, les algorithmes sont généralement fragiles à des transformations visuelles (par exemple, ré-encodage, changement du taux de résolution/bits). Par exemple, les données cachées sur un film seront probablement perdus lorsque le clip est compressé et transféré à sur

Introduction Générale

site Web. Comme solution complémentaire au tatouage numérique, le mécanisme de détection de copie vidéo est apparu. Par opposition au watermarking, qui repose sur l'insertion d'un motif distinct dans le flux vidéo, les techniques de détection de copie vidéo se basent sur l'extraction des caractéristiques du contenu pour détecter des copies de la vidéo. Plusieurs schémas de détection de copie typique basés sur le contenu spatial des images de la séquence vidéo ont été proposés dans la littérature. Dans cette mémoire, nous nous sommes intéressés à l'application du descripteur LBP et ses variantes pour la description du contenu vidéo. La motivation derrière l'utilisation de ces descripteurs vient de leur simplicité et leur application réussie dans divers domaines de traitement d'images tel que les systèmes de détection de reconnaissance visage.

Notre première contribution dans le cadre de cette thèse consiste à tester tous les descripteurs LBP et déduire la variante du descripteur qui est robuste au maximum de manipulations de traitements d'images. L'objectif de notre deuxième contribution est d'améliorer les performances du système réalisé.

Le présent document est structuré en quatre chapitres :

Le chapitre 1 est consacré à la présentation des principaux concepts, outils et travaux relatifs au système de détection de copie vidéo par le contenu

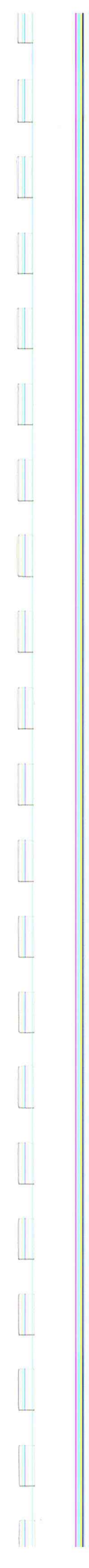
Le chapitre 2 est dédié à une étude générale des descripteurs utilisés dans les systèmes de détection de visage afin de déterminer les points forts et les points faibles de chacun. Puis nous nous sommes concentrées sur l'étude détaillée du descripteur LBP et ses extensions. Et enfin nous terminons par un bref état de l'art des principaux travaux de recherche élaborés pour la détection de copie vidéo basé sur le contenu.

Du chapitre 3, se détache des aspects théoriques abordés dans les deux premiers chapitres et s'oriente vers la présentation de notre contribution principale qui consiste à appliquer les descripteurs locaux LBP et ses variantes selon plusieurs voisinages. Dans ce chapitre, une comparaison des performances des descripteurs LBP en termes de robustesse par rapport aux attaques est effectuée pour en déduire la variante de LBP qui est robuste au maximum d'attaques. De là découle, l'application du descripteur spatio-temporel LBP-TOP uniforme avec un nombre de voisin égal à 8 dans le système de détection de copie vidéo. Une amélioration de la robustesse de ce descripteur par rapport à l'attaque de rotation est effectuée. Celle-ci est réalisée par l'introduction de l'algorithme log polaire a été effectuée

Introduction Générale

pour rendre le descripteur spatio-temporel plus robuste à l'attaque de rotation. Nous terminons le chapitre par une comparaison de notre travail avec les plates formes de détection de copie vidéo réalisées au CDTA.

A la fin de ce mémoire, nous émettons nos conclusions sur le logiciel de détection de copie vidéo basé sur le contenu développé. Et Afin d'améliorer le travail réalisé, nous envisageons plusieurs perspectives.



Chapitre I

Généralité sur le système de détection de copie vidéo

Introduction

Le récent essor des moyens de production et de diffusion des contenus multimédia entraîne la création et l'échange d'une très grande quantité de programmes. La structuration automatique de ces grands volumes de données par la fouille du contenu non textuel devient indispensable et constitue un défi majeur pour les prochaines années [1].

Pour des raisons essentiellement culturelles, le format court domine actuellement dans la nature des programmes audiovisuels produits (séries, reportages courts, clips musicaux, etc.) et dans la façon de les réaliser (plans courts et rapides, montages différents des mêmes séquences, etc.). Ces contenus font souvent l'objet de rediffusions ou sont réutilisés, en entier ou par fragments, dans d'autres programmes. Aussi, le grand public n'est plus un simple consommateur mais assume de plus en plus un rôle de producteur de contenus, bien souvent par le réagencement de contenus existants, parfois agrémentées de façon originale. En conséquence, les bases de programmes d'institutions comme l'INA (Institut National de l'Audiovisuel) ou de sites Web2.0 de partage de vidéos contiennent un nombre important de versions transformées des mêmes contenus de départ [1].

L'identification automatique, au sein d'une très grande base multimédia, des liens entre les différentes versions d'un même contenu est basée sur la détection de copies par le contenu (CBCD). Détection de copie de la vidéo est indispensable à de nombreuses applications du monde réel, y compris la protection du droit d'auteur, le droit enquêtes policières, business intelligent, annonce le suivi et l'élimination de la redondance [2].

Dans ce chapitre, une description de la structure et les différents composants et caractéristiques d'une vidéo sont présentés suivi d'un état de l'art sur le système de détection de copie vidéo.

I. Définition d'une vidéo

La vidéo est une succession d'images (trames) qui constitue l'évolution temporelle d'une scène. Ces images sont défilées à une certaine cadence afin de créer une illusion de mouvement pour l'œil humain. Elle peut être analogique (signal continu d'intensité de luminance) ou numérique (suite de trames ou images) [3]. La figure (I.1) représente les composants de la vidéo qui sont:

- **Scène** : par définition elle regroupe l'ensemble des plans consécutifs, une scène décrit un événement de manière continue. Elle possède deux caractéristiques qui sont l'unité de temps et l'unité d'action [4].
- **Plan** : C'est une prise de vue de la caméra, un morceau d'enregistrement effectué sans interruption ayant un début et une fin. Ainsi, il est défini comme une séquence d'image durant laquelle l'acquisition du signal n'a pas subi d'interruption [5].
- **Image** : est une matrice de pixels repérés par leur coordonnées (x, y). S'il s'agit d'une image couleur, un pixel est codé sur trois composantes (R, V, B). S'il s'agit d'une image en niveau de gris, le pixel est codé par une seule composante [6].

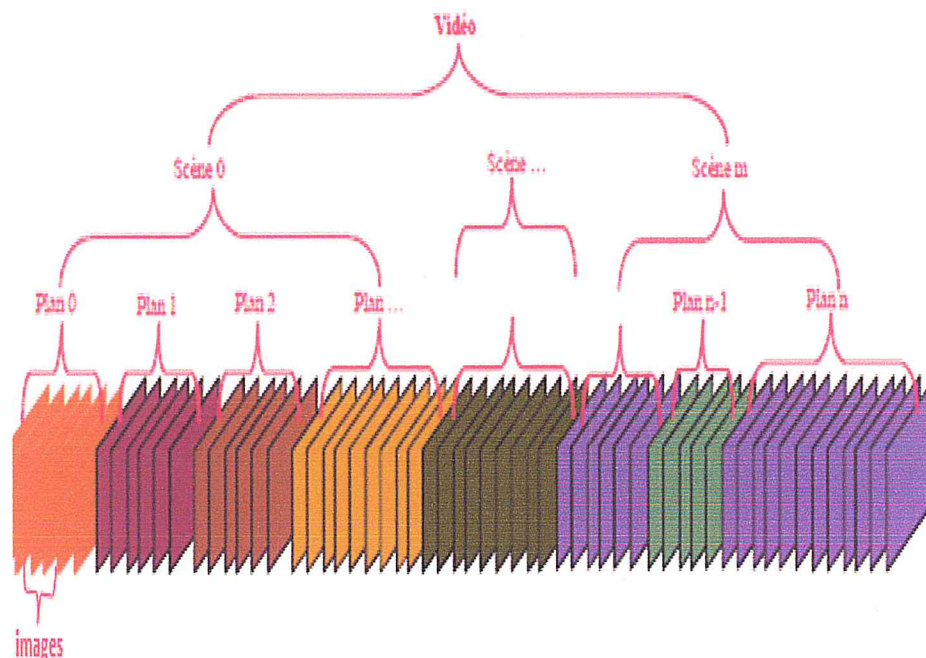


Figure I. 1: Les composantes d'une vidéo [6].

I.1 Caractéristique d'une vidéo

Les caractéristiques d'une vidéo les plus couramment utilisées sont :

I.1 .1 La couleur

La couleur est une caractéristique riche d'information et très utilisée pour la représentation de l'image et de la vidéo. Elle forme une partie significative de la vision humaine. Le codage de cet attribut dans les images peut être effectué par l'identification de l'espace couleur le plus discriminant [7]. Nous citons les principaux espaces:

➤ L'espace RVB (Rouge Vert Bleu) ou RGB (Red Green Blue)

Dans cet espace, chaque couleur est définie par une combinaison de trois couleurs primaires (R, V, B). A partir de ces trois couleurs, il est possible de faire leur trois vecteurs $(\vec{R}, \vec{V}, \vec{B})$ correspondant afin de former un espace vectoriel d'origine O. Dans cet espace, chaque stimulus de couleur peut être représenté par un point C de coordonnées (R_C, V_C, B_C) . Les points correspondant à des stimuli de couleur dont les niveaux de couleur sont positifs sont contenus dans un cube, connu sous le nom de cube de couleurs (figure I.2). L'origine $O(0,0,0)$ correspond à la couleur noire tandis que la couleur blanche de référence correspond au point de coordonnées $(1,1,1)$ qui est le mélange des trois couleurs (R,V,B) [8].

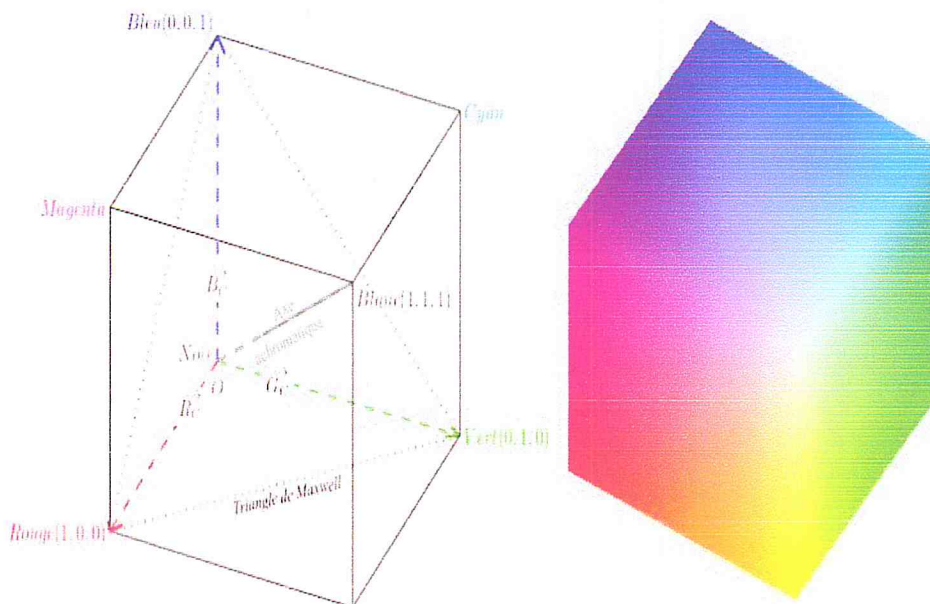


Figure I. 2: Le cube de couleur [8].

Généralités sur le système de détection de copie vidéo

➤ L'espace XYZ

Cet espace prend en compte la sensibilité de l'œil, dont les primaires (X, Y, Z) sont virtuelles. La composante Y représente la luminance, X et Z contiennent l'information de chrominance. La figure I.3 représente le diagramme de cet espace [9].

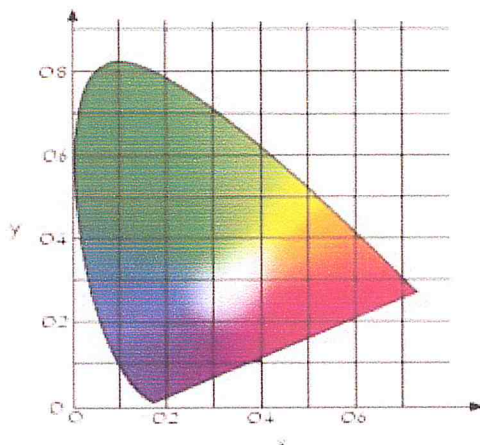


Figure I. 3: Représentation de l'espace XYZ [9].

➤ L'espace YUV

Dans cet espace La composante Y est commune à l'espace XYZ, elle représente donc toujours la luminance, tandis que les composantes U et V contiennent les informations de chrominance [9].

➤ Espace Hue, Saturation, Intensity (HSI)

Le principe de cet espace est de caractériser les couleurs de façon plus intuitive et de se rapprocher de la perception humaine. Les trois composantes de cet espace sont la teinte, la saturation et l'intensité. Le cône de la figure I.4 illustre un modèle de cet espace. Dans ce modèle, la teinte est un angle de 0 degrés à 360 degrés, elle représente la couleur pur. La saturation indique la gamme de gris dans l'espace couleur. Elle varie de 0 à 100%. L'intensité est la mesure de la luminosité de la couleur, qui doit varier entre la noire et le blanc [10].

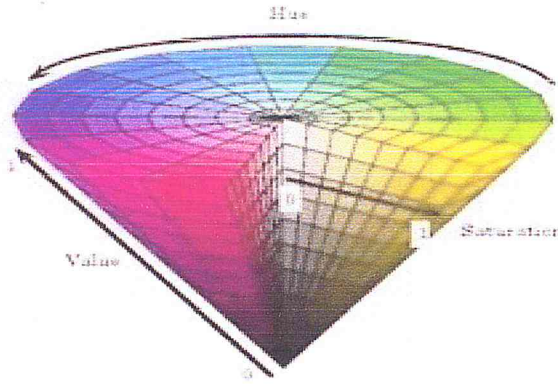


Figure I. 4: Représentation de l'espace HSI [10].

I.1.2 La texture

La texture est le second attribut visuel qui représente globalement ou partiellement une image [10]. La définition littéraire de la texture est la répétition spatiale d'un même motif dans différentes directions de l'espace. Cette définition est limitative car elle caractérise l'objet indépendamment d'un observateur humain. La notion de texture est utilisée pour traduire un aspect homogène de la surface d'un objet contenu dans une image [11]. Dans la pratique, les auteurs définissent la texture selon deux aspects qui correspondent à deux niveaux de perception.

- les macrotextures** : présentent un aspect régulier, sous formes de motifs répétitifs spatialement placés selon une règle précise et suivant une approche structurale déterministe [12].
- les microtextures** : présentent des primitives "microscopiques" distribuées de manière aléatoire suivant une approche probabiliste cherchant à caractériser l'aspect anarchique et homogène [13].

I.1.2.1 Classification de la texture

La complexité de la notion de texture s'impose aussi dans la classification de l'ensemble des textures, plusieurs catégories existent. Citons les trois principales familles [14] [15].

- **Textures structurales** : la texture de cette classe est considérée comme étant la répartition spatiale de motifs élémentaires de base dans différentes direction de l'espace suivant une certaine règle de placement (figure I.5) [14].

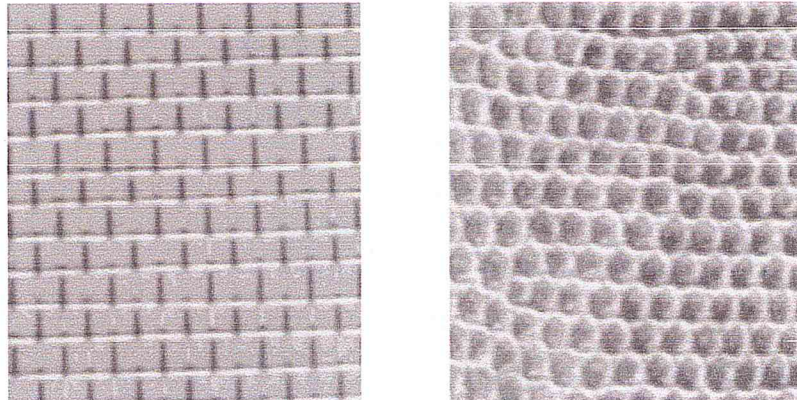


Figure I. 5:Textures structurées [14].

➤ **Textures aléatoires :** Ce type de texture a un aspect anarchique tout en restant globalement homogène. Le motif de base se répète spatialement, il ne peut être extrait [15]. Dans cette catégorie, la texture est considérée comme étant un processus aléatoire bidimensionnel (figure I.6).

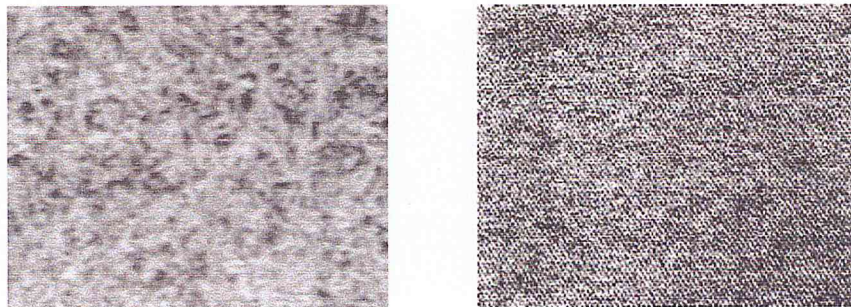


Figure I. 6: Textures aléatoires [15].

➤ **Textures directionnelles :** Ces textures ne sont pas totalement aléatoires et ne possèdent pas d'éléments structurants de base [15]. Néanmoins, elles se caractérisent par certaines orientations. La figure I.7 illustre ces textures. La texture de gauche laisse apparaître des lignes obliques, tandis que celle de droite possède des lignes verticales.

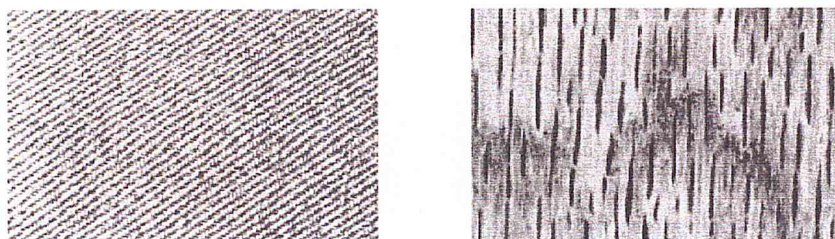


Figure I. 7: Textures directionnelles [15].

Généralités sur le système de détection de copie vidéo

I.1.3 La forme

Les attributs de forme sont complémentaires de celle de couleur, pour décrire la structure générique du contenu. Elle est généralement une description très riche d'un objet. L'extraction d'attribut géométrique a été le fer de lance la recherche d'image par contenu [10] [16].

I.1.4 Le mouvement

Contrairement aux trois attributs cités précédemment, l'attribut mouvement est spécifique à la vidéo. C'est une composante riche en information, elle renseigne sur l'activité d'un plan et celle de ses objets. Son analyse consiste généralement à appairer des composants d'une image à un instant donné avec les composantes des images suivantes ou précédentes, l'écart de position spatial représente un vecteur mouvement caractérisant la direction et l'amplitude [17].

I.2 La résolution d'une image

La résolution d'une image est définie par un nombre de pixels par unité de longueur de la structure à numériser classiquement en ppp (points par puce). Ce paramètre est défini lors du processus de numérisation et dépend principalement des caractéristiques du matériel. Plus le nombre de pixels est élevé par unité de longueur de la structure à numériser, plus la qualité d'information qui décrit cette structure est importante et plus la résolution est élevée. La résolution d'une image numérique définit le degré de détail qui va être représenté sur cette image [18].

I.3 Résolution temporelle d'une vidéo

La résolution temporelle d'une vidéo est définie par le nombre d'images défilant par seconde. Afin d'éviter les désagréments dus aux papillotements, et prenant en considération les spécificités de l'œil humain qui garde une image environ 10 micro secondes, les images doivent défiler à une certaine cadence [3]:

- ✓ environ 24 images/s pour un film de cinéma.
- ✓ 25 images/s pour la télévision européenne.
- ✓ environ 30 images/s pour la télévision nord-américaine et japonaise.

Généralités sur le système de détection de copie vidéo

I.4 Les différents formats de fichiers vidéo

Il existe plusieurs formats de fichiers vidéo, les formats les plus courants sont résumés dans le tableau I.1 [19].

Format	Caractéristique
MOV (Quick Time)	Un format de codage de données vidéo et audio créé par Apple. Il est l'un des plus anciens formats de compression vidéo. il peut contenir des pistes audio, vidéo, et des textes. Nécessite le lecteur Quicktime.
AVI (Audio Video Interleave)	Un format de fichier vidéo créé par Microsoft. Ce format prévoit une définition d'image maximale de 320 x 240 pixels et une cadence maximale de 30 images/seconde. Ce format est très répandu, il peut contenir tout type de fichier audio et vidéo, mais pas de textes. Ce format fonctionne sur n'importe quelle machine, et ne nécessite pas de carte d'extension particulière.
MPEG (Motion Picture Experts Group)	un format d'organisme de standardisation américain du secteur de la vidéo et de l'audio, est un format de fichier audio et vidéo utilisant la compression avec pertes. il est utilisé pour la diffusion en continu sur Internet (le streaming). Il est compatible avec tous les logiciels et les platines DVD. Il existe plusieurs standards de MPEG comme MPEG-1, MPEG-2 et MPEG-4.
RA (Real Média)	Format développé par Real Networks, il supporte de nombreux formats (spécialement les formats Real Audio et Real Movie). Il est adapté au streaming, la lecture est uniquement avec Real Player.
WMF Windows Média File	Il est le format de codage de données multimédias créé par Microsoft. WMF est le premier format à assurer la gestion des droits sur la séquence audio ou vidéo.
MP4	Il est très utilisé sur Internet, souvent utilisé pour les DivX. Très souple, il permet l'affichage d'un flux vidéo au cours de sa diffusion. Pour lire une vidéo enregistrée dans le format MP4, il suffit d'utiliser un lecteur multimédia.

Tableau I. 1: Les principales formats de fichiers vidéo.

Généralités sur le système de détection de copie vidéo

I.5 Une copie vidéo

Une vidéo V1, au moyen de diverses transformations telles que l'addition, redimensionnement suppression, modification (la couleur, le contraste, l'encodage, flou, recadrage, ajouter des logos ou des cadres, redimensionner etc.), l'enregistrement par caméscope, et ainsi de suite, se transforme en une autre vidéo V2 qui est perçue comme étant très similaire à V1, puis la vidéo V2 est appelé une copie de la vidéo V1 [10]. La figure I.8 représente un exemple d'une copie vidéo telle que la figure de gauche est une image de vidéo d'origine, tandis que celle de droite est une image d'une vidéo copie.



Figure I. 8 : Exemple d'une copie vidéo [10].

II. Système de détection de copie vidéo

La détection de copie vidéo est essentielle pour de nombreuses applications, et peut être utilisée à plusieurs usages. Elle permet de retrouver simultanément toutes les versions d'une même vidéo, donc la détection de copie vidéo est un domaine de recherche actif en raison de la nécessité d'une protection du droit d'auteur et des affaires de protection de la propriété intellectuelle [20]. Il existe deux approches pour détecter les copies d'un contenu vidéo:

- Détection de copie vidéo par tatouage numérique (watermarking).
- Détection de Copie basé sur le Contenu (CBCD).

II.1 Tatouage numérique (watermarking)

Le tatouage numérique consiste à insérer une signature dans un contenu vidéo de manière invisible. Ce dernier modifie le contenu dans le domaine spatial ou dans le domaine

Généralités sur le système de détection de copie vidéo

fréquentiel de façon à ne pas altérer la qualité de la vidéo. Néanmoins, ce mécanisme présente des inconvénients à savoir :

- Le processus d'insertion doit s'effectuer avant que les vidéos sont distribuées ;
- Les informations insérées peuvent être facilement détruites par les attaques malveillantes et peuvent être endommagés après un certain traitement du contenu.

La raison pour laquelle, un autre mécanisme alternative à la technologie de tatouage numérique a été élaboré. Il s'agit de la technique de détection de copies vidéo basée sur le contenu (CBCD), axe de recherche ans lequel notre travail s'insère.

II.2 Détection de Copie Basé sur le Contenu (CBCD)

La détection de copies vidéos basée sur le contenu (ou CBCD : « Content-Based Copy Detection ») est une sorte de recherche des vidéo par le contenu (CBVR : «Content-Based Video Retrieval» qui est largement utilisée pour récupérer les répliques d'une vidéo donnée (vidéo requête) à partir d'une grande collection de vidéos. Le CBCD et le CBVR ont la même architecture, néanmoins, il existe une différence dans l'objectif poursuivi. Le CBCD sert à détecter les vidéos copies d'une vidéo requête, alors qu'un système CBVR sert à trouver des vidéos qui ont quelques éléments similaires à ceux de la vidéo requête, ces éléments étant décrits au préalable (couleur, texture, forme, etc.). La détection de copie peut ainsi être traitée comme un cas restreint de la recherche par le contenu. Un système CBCD peut être utilisé pour trouver des copies illégales d'une vidéo donnée. Ces copies peuvent avoir subi quelques déformations (rotations, changement de contraste, compressions, redimensionnement, ..) sans pour autant affecter l'interprétation de la vidéo ou son contenu sémantique. Le but de ces déformations est justement d'échapper aux détecteurs automatiques de copies [20] .

II.2.1 Les étapes de la détection de copie basée sur le contenu (CBCD)

Le système de détection de copie vidéo à base de contenu comprend deux étapes principales. La première c'est une étape hors ligne (Off-line) et la deuxième c'est l'étape en ligne (On-line) (figure I.9) [20].

Généralités sur le système de détection de copie vidéo

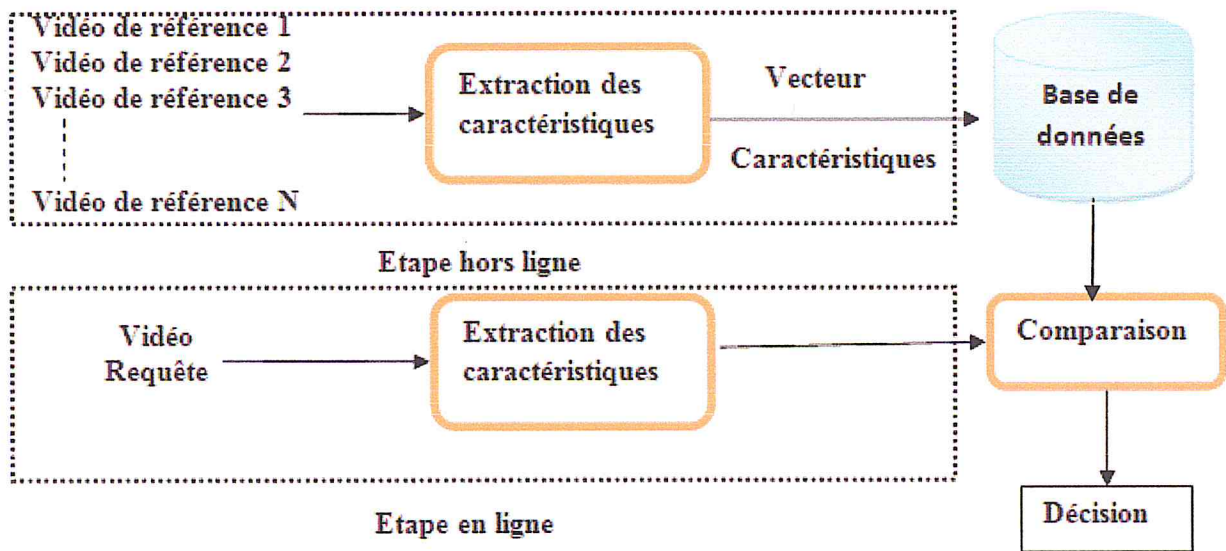


Figure I. 9: Les étapes de détection de copie vidéo basée sur le contenu (CBCD) [20].

➤ L'étape hors ligne (Off-line)

Comme dans des nombreux systèmes de recherche basés sur le contenu, les vidéos de référence sont indexées dans une étape hors ligne. Dans cette étape, les caractéristiques de chaque vidéo de référence sont extraites et sont mises dans un vecteur qu'on appelle descripteur de caractéristiques. Ce dernier est stocké dans une base de données.

➤ L'étape en ligne (On-line)

De manière similaire que l'étape hors ligne, les caractéristiques de la vidéo requête (cible) sont extraites et mises dans un descripteur, qui est ensuite comparé aux descripteurs stockés dans la base de données. La comparaison est effectuée selon des métriques de similarité.

II.2.2 La similarité

En général, la similarité est une fonction qui quantifie le rapport entre deux objets comparés en fonction des points de ressemblance et de différence. Bien entendu, ces deux objets doivent appartenir à une même classe sémantique.

La similarité entre deux documents vidéo, telle qu'elle est vue par *Cheung et al* [21] [22] est donnée comme étant le pourcentage d'images ou de plans semblables partagés par ces documents. Cette mesure est semblable à celle appliquée aux documents textuels [23], mesure qui consiste à calculer le pourcentage des mots semblables partagés. Cependant, pour une vidéo, cette mesure est plus délicate à mettre en œuvre. Pour mesurer la similitude des images

Généralités sur le système de détection de copie vidéo

- $d(j, i) = d(i, j)$ (symétrique)
- $d(i, j) = 0 \leftrightarrow i = j$ (identité)
- $d(i, j) \leq d(i, k) + d(k, j)$ (inégalité triangulaire)

Si toutes les propriétés citées auparavant sont respectées, on se trouve en présence d'une distance métrique. Si l'inégalité triangulaire n'est pas respectée, on parle plutôt d'une distance semi- métrique.

Dans la littérature de nombreuses distances vérifiant les propriétés précédentes sont publiées. Aucune d'elle ne peut être considérée comme optimale pour une classification. Cela dépend des images à traiter et des algorithmes de classification mis en oeuvre [30].

1. les distances dans l'espace d'image

Soient deux vecteur $x (x_1, \dots, x_n)$ et $y (y_1, \dots, y_n)$ de l'espace n dimension. La distance entre ces vecteurs peut s'exprimer de différentes façons dont les plus courantes sont indiquées ci-dessous [10]:

1.1 Distance de MinKowsky definit par :

$$d(x, y) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{1/p} \quad (1.1)$$

Où $p \geq 1$ est le facteur de Minkowsky. Selon les valeurs de p cette distance définit d'autres distances qui sont I.2 :

Valeur de p	Nom de la distance	Formules mathématiques
1	Distance de Manhattan (city-block)	$d(x, y) = \sum_{i=1}^n x_i - y_i $
2	Distance Euclidienne	$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$
Lorsque p tend vers l'infini	Distance de Chebychev, aussi appelée distance « Queen-wise » ou encore métrique maximum:	$d(x, y) = \max_{1 \leq i \leq n} x_i - y_i $

Tableau I. 2: Les distances de MinKowsky.

1.2 Distance se Chi square χ^2

$$d(x, y) = \sum_{i=1}^n \frac{(x_i - y_i)^2}{(x_i + y_i)} \quad (1.2)$$

Dans certains articles, la formule de la distance de Chi Square est:

$$d(x, y) = \frac{1}{2} \sum_{i=1}^n \frac{(x_i - y_i)^2}{(x_i + y_i)} \quad (1.3)$$

2. Distances dans l'Espace de Mahalanobis

Avant de pouvoir effectuer des mesures de distance dans l'espace de Mahalanobis [31], il est essentiel de comprendre comment l'on passe de l'espace des images Im à l'espace de Mahalanobis (EMah).

2.1 De l'espace des images à l'espace de Mahalanobis

L'espace de Mahalanobis est un espace où la variance selon chaque dimension est égale à 1. On l'obtient à partir de l'espace des images Im en divisant chaque vecteur propre par son écart-type correspondant.

Soit u et v deux vecteurs propres de Im , et m et n deux vecteurs de EMah. Soit λ_i les valeurs propres associées aux vecteurs u et v , et σ_i l'écart type, alors on définit $\lambda_i = \sigma_i^2$. Les vecteurs u et v sont reliés aux vecteurs m et n à partir des relations suivantes :

$$m_i = \frac{u_i}{\sigma_i} = \frac{u_i}{\sqrt{\lambda_i}} \quad \text{et} \quad n_i = \frac{v_i}{\sigma_i} = \frac{v_i}{\sqrt{\lambda_i}} \quad (1.4)$$

2.2 Mahalanobis L1 (MahL1)

Cette distance est exactement la même que la distance City-Block sauf que les vecteurs sont projetés dans l'espace de Mahalanobis. Ainsi, pour des vecteurs propres u et v de projection respectives m et n sur l'espace de Mahalanobis, la distance Mahalanobis $L1$ est définie par:

$$\text{MahL1}(u, v) = \sum_{i=1}^n |m_i - n_i| \quad (1.5)$$

2.3 Mahalanobis L2 (MahL2)

Cette distance est identique à la distance euclidienne à part qu'elle est calculée dans l'espace de Mahalanobis. Ainsi, pour des vecteurs propres u et v de projection respectives m et n sur l'espace de Mahalanobis, la distance Mahalanobis L2 est définie par:

$$\text{MahL2}(u, v) = \sqrt{\sum_{i=1}^n |m_i - n_i|^2} \quad (1.6)$$

2.4 Cosinus de Mahalanobis (MahCosine)

Il s'agit tout simplement du cosinus de l'angle entre les vecteurs u et v , une fois qu'ils ont été projetés sur EMah et normalisés par des estimateurs de la variance.

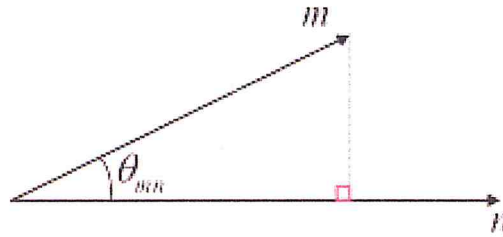


Figure I. 10: deux vecteurs m et n dans l'espace de Mahalanobis.

Par définition [31]:

$$\text{Smahcosince}(u, v) = \cos(\theta_{mn}) \quad (1.7)$$

De plus, on peut écrire :

$$\cos(\theta_{mn}) = \frac{|m||n| * \cos(\theta_{mn})}{|m||n|} \quad (1.8)$$

D'où la formule finale de la mesure de similarité MahCosine

$$\text{Smahcosince}(u, v) = \frac{m \cdot n}{|m||n|}, \text{Dmahcosince}(u, v) = -\text{Smahcosince}(u, v) \quad (1.9)$$

II.2.3 Classifications des méthodes de CBCD

Dans la littérature, plusieurs méthodes de CBCD ont été proposées [32], ils peuvent être regroupés en fonction du type de caractéristiques extraites. Les méthodes CBCD peuvent être classifiées selon le schéma illustré sur la figure I.II.

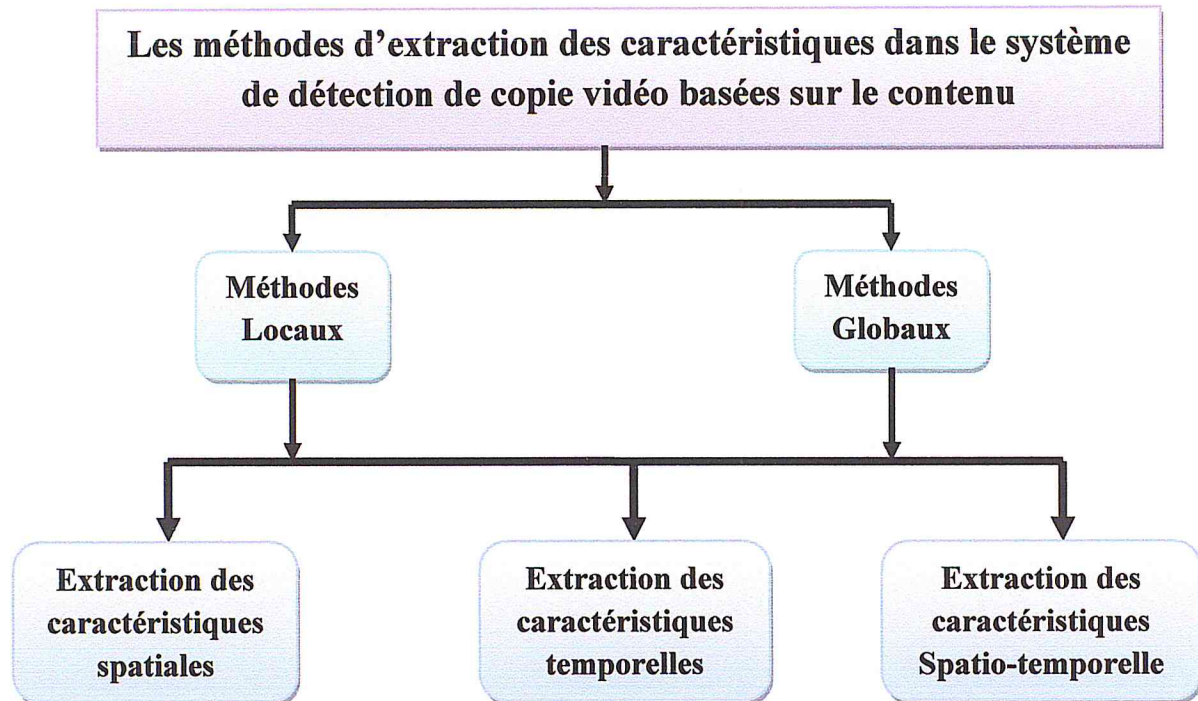


Figure I. 11: Les méthodes de détection de copie vidéo basée sur le contenu (CBCD).

II.2.3.1 Méthodes Globales vs Locales

Les méthodes peuvent être classées en méthodes globales ou locales selon la nature du descripteur global ou local [33]. Le tableau I.2 récapitule les principales caractéristiques de ces descripteurs.

Descripteur local	Descripteur global
<ul style="list-style-type: none">➤ Utiliser le critère de localité en basant sur l'extraction des points d'intérêt ou des régions locales.➤ Subdivision de l'image en blocs	<ul style="list-style-type: none">➤ utiliser les caractéristiques globales de l'image comme la couleur ou la texture des histogrammes.➤ Travailler sur l'image entière.

Tableau I. 3: Caractéristiques de descripteurs locaux et globaux.

Les caractéristiques globales contiennent les propriétés globales d'une image (par exemple, histogrammes d'image), tandis que les caractéristiques locales représentent habituellement l'information locale autour des points d'intérêt dans une image (par exemple, des bords, coins, etc.).

Généralités sur le système de détection de copie vidéo

Pour accélérer le processus d'extraction des caractéristiques dans le système de détection de copie vidéo, d'autres techniques ont été proposées [34]. Celles-ci sont basées sur les images clés de la séquence vidéo (Key frame). Ces images clés sont définies par l'activité visuelle globale et apparaissent à un rythme moyen par seconde, typiquement lors de changements de plan. La succession des images clés peut être vue comme un résumé précis d'une vidéo. Donc l'extraction des caractéristiques d'un ensemble d'images clés est faite après la détection des images clés au lieu de chaque image de la séquence vidéo. La figure I.12 représente l'extraction des caractéristiques basées sur les images clés.

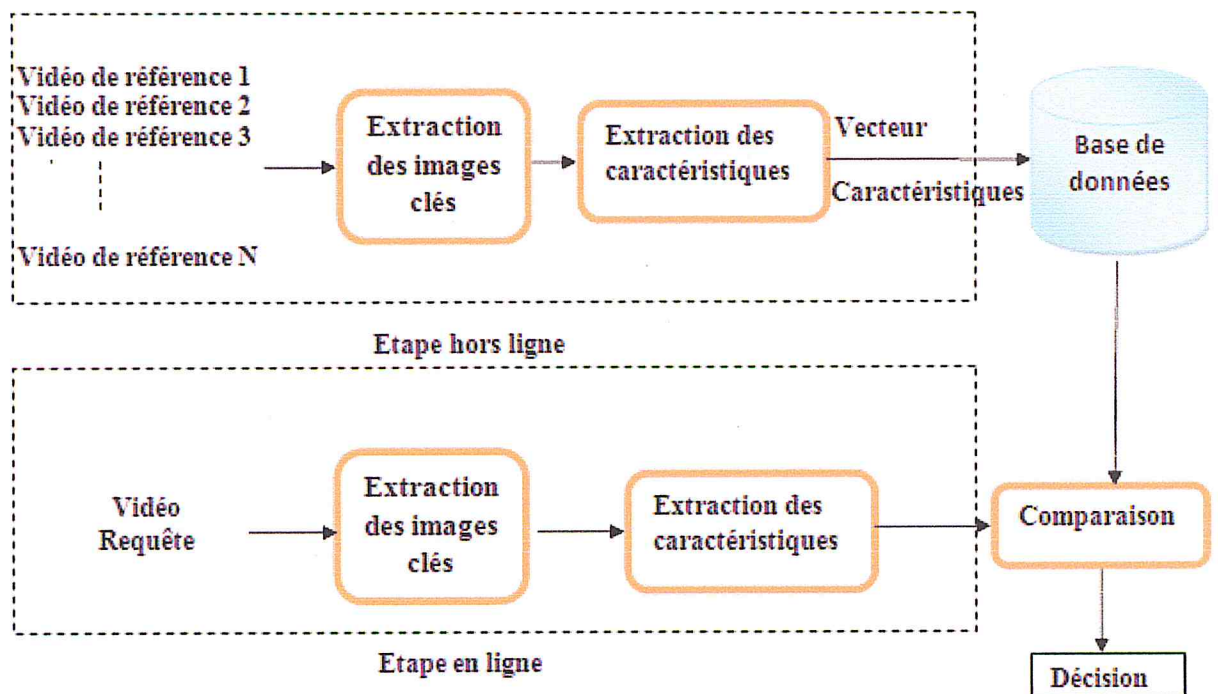


Figure I. 12: Détection de copie vidéo basée sur les images clés [20].

II.2.3.1.1 Extraction des caractéristiques spatiales

L'extraction des caractéristiques spatiales peuvent être classifiée en deux classes :

➤ Les méthodes basées sur les mesures ordinales

L'une des premières méthodes CBCD basée sur les caractéristiques spatiales a été proposée par *Bhatt et al* [35]. La mesure ordinale de chaque image représente la moyenne des intensités de niveaux de gris. Les images en entrée subissent une subdivision en blocs de taille N ($N=N_x \times N_y$). Pour chaque bloc, l'intensité moyenne de niveau de gris est calculée. Les résultats de chaque région sont classés pour produire un vecteur de dimension

Généralités sur le système de détection de copie vidéo

$S_N(t) = (r_1, r_2, \dots, r_N)$ pour chaque image clé.

➤ Les méthodes basées sur l'histogramme de couleur

Ces méthodes sont des méthodes globales, principalement obtenues à partir des histogrammes des couleurs dans des régions spécifiques comme s'est illustré dans la figure I.13.

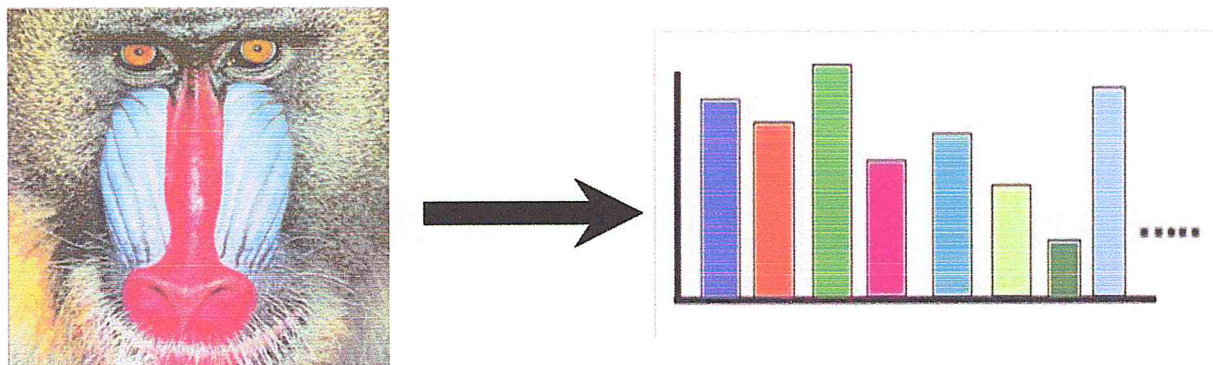


Figure I. 13:Histogramme de couleur [36].

Naphade et al. [36] ont proposé un système qui utilise l'histogramme dans l'espace de couleur YUV . La méthode repose sur la concaténation de l'histogramme de luminosité Y et les deux histogrammes de chrominance UV . L'histogramme résultant constitue la signature de la vidéo. La comparaison des signatures issues des vidéos d'origines et celle de la vidéo requête se base sur l'intersection des histogrammes. L'histogramme YUV est calculé pour chaque image de la séquence vidéo. Il utilise 32 classes pour représenter la composante Y , et 16 classes pour chacune des composantes U et V . Il en résulte une signature basée sur l'histogramme concaténé par 64 classes.

Le principe inconvénient des méthodes spatiales est leur incapacité à incorporer l'information temporelle de la vidéo, ce qui est un facteur discriminant importante. D'autres parts, les caractéristiques de couleur ne sont pas applicables à des vidéos en noir et blanc.

II.2.3.1.2 les méthodes basées sur les caractéristiques temporelles

Un certain nombre de techniques basées sur la caractéristique temporelle des séquences vidéo ont été proposées. *Indyk et al.* [37] [38] ont proposé une méthode dont le principe est comme suit : La vidéo est divisée en images clés. Ensuite la durée de chaque image clé est prise comme une signature temporelle, la séquence de durées concaténées forme l'empreinte digitale de la vidéo. Le principe inconvénient de cette méthode est que la méthode opère

Généralités sur le système de détection de copie vidéo

uniquement sur de longues séquences vidéo. Ceci car les séquences vidéo de courte durée ne contiennent pas d'information temporelle suffisamment discriminante.

II.2.3.1.3 Les méthodes basées sur les caractéristiques spatio-temporelles

Les techniques basées sur les caractéristiques spatio-temporelles combinent les caractéristiques spatiales et temporelles.

Malekesmaeili et al. [39] ont proposé une technique basée sur la construction d'un descripteur formé à partir des caractéristiques spatiales et temporelles TIRIs (temporally informative representative images). Le système TIRI s'applique également sur des séquences vidéo de courte durée. La caractéristique spatiale extraite du système TIRI contient également des informations temporelles. Ceci est fait simplement en mettant les images de la vidéo dans un intervalle de temps donné (T).

Simand et al. [40] ont proposé une fonction d'extraire la signature spatio-temporelle pour représenter les vidéos et construire une structure de données efficace en temps réels. Cette fonction exploite la distribution d'intensité de niveau de gris de toute la séquence d'images.

Geert Willems *et al.* [41] ont proposé la génération d'un descripteur de la vidéo basé sur l'extraction des points d'intérêt spatio-temporel.

Deux autres méthodes développées au CDTA (centre de développement de Technologies Avancées) ont été proposées [40] [41]. Les deux méthodes reposent sur l'utilisation d'un descripteur local « Binarized Statistical Image Features » (BSIF) et l'intensité moyenne. L'approche publiée dans [40] est robuste aux différentes transformations à l'exception de rotation. Les auteurs ont remédié à cet inconvénient, en proposant la segmentation de l'image en anneau au lieu de la segmentation en blocs [42].

Conclusion

Dans ce premier chapitre, nous avons présenté de manière générale les notions de base d'une vidéo, suivie par la description générale des systèmes de CBCD et leur classification. Un bref état de l'art des approches proposées dans la littérature est présenté à la fin de ce chapitre. Dans le prochain chapitre, on présentera les différents descripteurs utilisés dans les systèmes de détection de visage.



Chapitre II

Les descripteurs Locaux

Introduction

Un espace largement étudié de traitement d'image se concentre sur l'utilisation du contenu visuel pour la description de l'image. Ceci car le contenu visuel est très riche en informations qui permettent de bien caractériser l'objet photographié. Etant donné le nombre important de ces informations, les systèmes informatiques ne peuvent pas utiliser directement le contenu brut de l'image pour décrire, comparer ou identifier les objets qu'elle contient. D'où, il est nécessaire d'extraire une représentation compacte et discriminante qui permet de décrire les principaux aspects représentés par l'image. Il s'agit d'un ensemble de descripteurs de forme, de couleurs et de texture qui seront sauvegardés dans une base de données. Les recherches et les traitements ultérieures, seront faites selon cette représentation compacte et non pas selon le contenu brut des images [43].

Un descripteur peut être soit local, soit global. Un descripteur global utilise les caractéristiques visuelles de toute l'image, tandis qu'un descripteur local utilise les caractéristiques des régions ou des objets pour décrire le contenu de l'image. Il a été mentionné dans la littérature [44], que la caractérisation globale des images n'offre pas toujours des résultats satisfaisants, en particulier, dans les domaines où les caractérisations locales sont importantes tels que les systèmes de détection de visage et les systèmes de détection de copies vidéo.

Vue l'avantage de réduction du support de représentation des images en sous-ensembles significatifs robustesse et efficacité qu'offrent des descripteurs locaux, dans ce chapitre, nous nous intéressons seulement à ce type de descripteurs. Ce chapitre constitue une synthèse de quelques descripteurs locaux les plus utilisés dans les systèmes de reconnaissance et de détection de visage.

I. Le principe d'un descripteur local

Pour obtenir les caractéristiques visuelles locales, l'image en entrée est subdivisée en plusieurs parties. La méthode la plus simple est la subdivision de l'image en plusieurs blocs identiques en forme et en taille.

L'idée de base consiste à définir les caractéristiques locales de l'image (points ou régions). Ces descripteurs peuvent exploiter différentes propriétés de l'image comme la couleur de

pixel, la texture, des arêtes ou des coins. De nombreux descripteurs locaux ont été développés dans la littérature [43] [45] [46] [47]. Parmi ceux-ci, dans ce chapitre, les descripteurs les plus utilisés sont détaillés.

I.1 LPQ (Local Phase Quantization)

Le descripteur de texture LPQ a été introduit par *Ojansivu et al.* [48]. Il permet d'améliorer la classification de textures tout en étant robuste aux effets de blocks (artéfacts) générés par différentes formes de flou présents dans une image. Pour cela, le descripteur est construit de façon à ne retenir dans une image que l'information locale invariante à un certain type de flou.

L'information de LPQ est extraite en utilisant la transformée discrète de *Fourier* à fenêtre à deux dimensions (2D-WFT).

$$F_u(x) = \sum_{m \in Nx} h(m - x) f(m) e^{-j2\pi u^T m} = E_u^T f x \quad (2.1)$$

Où E_u est un vecteur de base de 2D-WFT, de taille $1 \times M^2$, u la fréquence et $f x$ est un vecteur de taille $M^2 \times N$ contenant les valeurs des pixels d'image dans Nx à chaque position x . La fonction fenêtre, $h(x)$ est une fonction rectangulaire. La transformation est calculée selon quatre valeurs de la fréquence $u = [u_0, u_1, u_2, u_3]$ où $u_0 = [a, 0]^T$, $u_1 = [0, a]^T$, $u_2 = [a, a]^T$ et $u_3 = [a, -a]^T$. La valeur a est la plus haute fréquence scalaire pour laquelle $h_{ui} > 0$. Ainsi, seules quatre fonctions complexes (comme un banc de filtres) sont nécessaires pour produire huit images résultantes, composées de 4 images de la partie réelle et 4 images de la partie imaginaire de la transformée. Chaque pixel de l'image complexe résultante peut être codé en une valeur binaire représentée dans l'équation suivante :

$$B_{u_i}^{Re}(x) = \begin{cases} 1 & \text{si } F_{u_i}^{Re}(x) > 0 \\ 0 & \text{si } F_{u_i}^{Re}(x) \leq 0 \end{cases}, \quad B_{u_i}^{Im}(x) = \begin{cases} 1 & \text{si } F_{u_i}^{Im}(x) > 0 \\ 0 & \text{si } F_{u_i}^{Im}(x) \leq 0 \end{cases} \quad (2.2)$$

En fait, LPQ fournit également la quantification de la fonction de phase de *Fourier*. En général, LPQ est une chaîne binaire obtenue pour chaque pixel par la concaténation des codes de bits réels et imaginaires des huit coefficients de Fourier u_i . LPQ est définie comme suit :

$$LPQ(x) = [B_{u_0}^{Re}(x), B_{u_0}^{Im}(x), \dots, B_{u_3}^{Re}(x), B_{u_3}^{Im}(x)] \quad (2.3)$$

La chaîne binaire est convertie en nombre décimal par l'expression (2.4) pour produire un histogramme de LPQ. La figure II.1 résume l'ensemble des étapes pour la génération du descripteur LPQ.

$$LPQ(x) = B_{u_0}^{Re}(x) + B_{u_0}^{Im}(x) \times 2^1 + \dots + B_{u_0}^{Re}(x) \times 2^{k-1} + B_{u_0}^{Im}(x) \times 2^k \quad (2.)$$

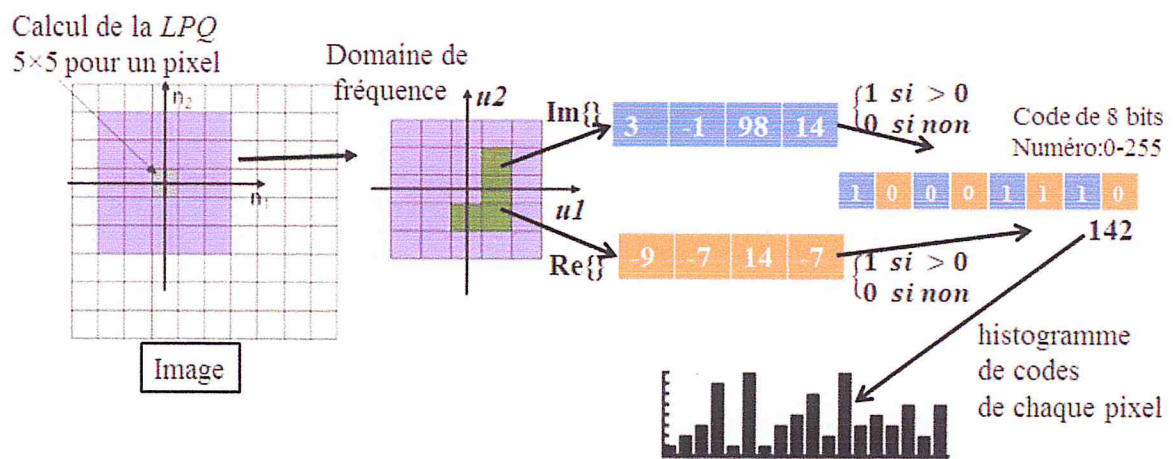


Figure II. 1: Organigramme de construction du descripteur LPQ [6].

I.2 BSIF (Binarized Statistical Image Features)

Contrairement au descripteur PLQ qui peut être utilisé pour calculer les statistiques d'étiquettes dans les voisinages des pixels locaux, le descripteur local appelé BSIF (Binarized Statistical Image Features), proposé par *Kannla and Rahtu* pour la reconnaissance faciale 2D et la classification de la texture, utilise un ensemble prédéfini de filtres linéaires et la binarisation des réponses des filtres [49].

Le principe du descripteur BSIF repose sur les points suivants :

Compte tenu d'une image patch X de taille $l \times l$ pixels et un filtre linéaire W_i de la même taille, la réponse du filtre si est obtenue par:

$$s_i = \sum_{u,v} w_i(u,v)X(u,v) = w_i^T x \quad (2.5)$$

Les descripteurs Locaux

Où les vecteurs w et x contiennent les pixels de W_i et X respectivement. La fonction binarisée b_i est obtenue par:

$$b_i = \begin{cases} 1 & \text{si } s_i > 0 \\ 0 & \text{autrement} \end{cases} \quad (2.6)$$

Les n filtres linéaires W_i , sont empilés sur une matrice W de taille $n \times l^2$. Le calcul de toutes les réponses à la fois est donné par $s = Wx$.

La chaîne de bits b_i est obtenue par la binarisation de chaque élément s_i de s . Ainsi, compte tenu des caractéristiques des filtres linéaires W_i , le calcul de b_i est simple. En outre, il est clair que les chaînes b_i pour tous les blocs de l'image de taille $l \times l$, entourant chaque pixel d'une image, peuvent être calculées par n convolutions.

Afin d'obtenir un ensemble utile de filtres W_i Kannla et Rahtu [49] ont estimé les filtres en maximisant l'indépendance statistique des s_i [50]. Les composantes indépendantes sont estimées en brisant la matrice de filtre dans deux parties par [8]:

$$s = Wx = UVx = Uz \quad (2.7)$$

Où $z = Vx$, et U est une matrice carrée de taille $n \times n$ qui sera estimée par ICA (Independent Component Analysis) et la matrice V effectue le prétraitement canonique, c'est le blanchiment et la réduction de la dimension simultanés des échantillons simples x [51].

Étant donné un ensemble d'images patches sélectionnés au hasard à partir d'images naturelles de l'ensemble d'apprentissage, les patches sont d'abord construits de moyenne nulle (l'intensité moyenne de chaque patch est soustraite) et leur dimension est réduite en ne conservant que les n premières composantes principales. Ces derniers sont divisés par leur écart-type pour obtenir des échantillons de données blanchis z . La décomposition en vecteurs propres et valeurs propres $C = EDE^T$ de la matrice de covariance C des échantillons x , la matrice V est définie par:

$$V = \left(D^{-\frac{1}{2}} E^T \right)_{1:n} \quad (2.8)$$

Les descripteurs Locaux

Où la diagonale principale de D contient les valeurs propres de la matrice C organisés dans l'ordre décroissant, et $(\cdot)_{1:n}$ désigne les n premières lignes de la matrice $(D^{\frac{1}{2}}E^T)$.

La matrice de filtres $W=UV$ obtenue, peut être utilisée directement pour le calcul du descripteur BSIF.

Il existe deux paramètres dans le descripteur BSIF: la taille du filtre l et la longueur n de la chaîne de bits.

- la taille des filtres peut être variable afin de capturer le contenu de l'image à de multiples résolutions : les filtres de grande taille sont appliqués pour gérer les basses fréquences et les effets de flou, alors que les filtres de petite taille sont appliqués pour capturer les variations de texture en haute fréquence.

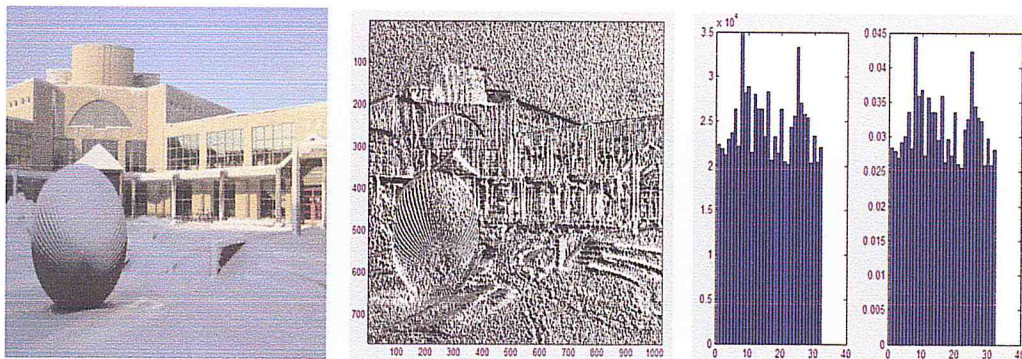


Image d'entrée

Image BSIF

Histogramme BSIF

Figure II. 2: Une image et un histogramme BSIF.

1.2.1 BSIF_TOP (Binarized Statistical Images Features Three Orthogonal Planes)

L'approche BSIF est étendue dans le domaine spatio-temporel en considérant une séquence d'images sur les trois plans orthogonaux (même principe que LBP-TOP) [10].

Où le YT et XT transmettent des informations sur les transitions espace-temps d'une séquence vidéo et le plan XY représente l'information spatiale.

En se basant sur le principe de BSIF, trois filtres différents sont utilisés sur les trois plans (XY , XT et YT). La normalisation puis la concaténation des trois histogrammes constitue les histogrammes spatio-temporel BSIF-TOP (figure II.3).

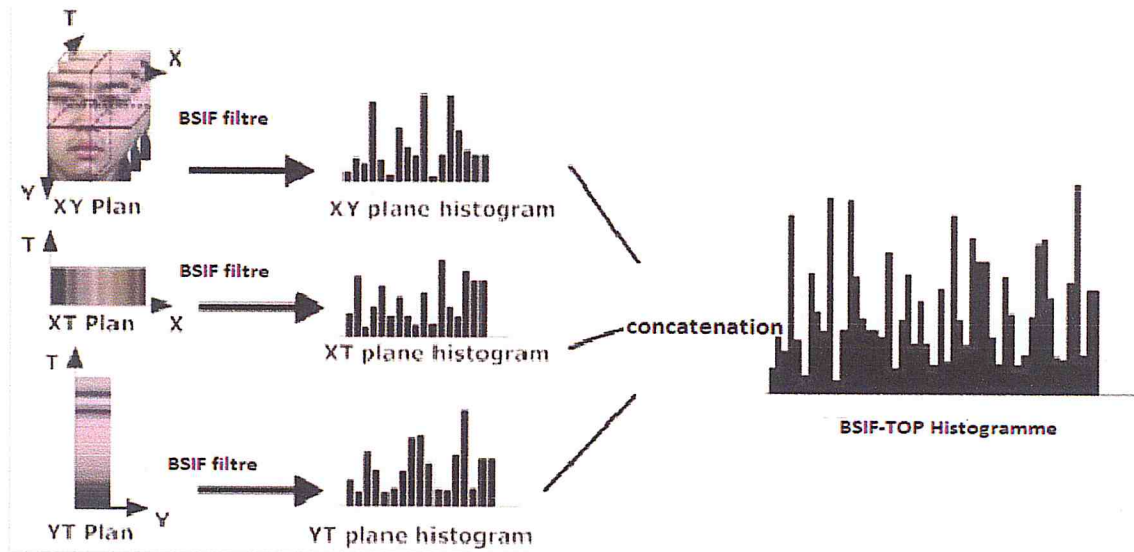


Figure II. 3: Principe de BSIF-TOP [10].

I.2.2 MBSIF (muti-scale BSIF)

Le descripteur MBSIF est une extension de BSIF-TOP multi-échelle pour capturer contenu dynamique à plusieurs résolutions [10]. BSIF-TOP à un cadre multi-échelle implique l'application de filtres de différentes tailles à une image. La construction de descripteur MBSIF peut être décrite comme suit:

L'application des opérateurs BSIF de différentes échelles (Z) à chaque image. Le résultat est un code de niveau de gris pour chaque pixel à chaque résolution. Une fois que le recadrage des images de codes résultants à la même taille, ils sont divisés en régions rectangulaires non chevauchantes $\{G_0, G_1, \dots, G_{J \times J - 1}\}$.

L'histogramme BSIF pour la région j dans l'échelle $h_{j,s}$ est calculé comme suit:

$$h_{j,s} = [h_{j,s}^0, h_{j,s}^1, \dots, h_{j,s}^{L-1}]$$

$$h_{j,s}^i = \sum_{pc \in G_j} 1\{BSIF_s(p_c) = i\} \quad (2.9)$$

$$j \in [0, 1, \dots, J \times J - 1]$$

$$s \in [1, 2, \dots, Z], L = 256$$

Où $1\{.\}$ est une fonction indicatrice si son argument est vrai / faux, et L représente le nombre de bins d'histogramme qui est égale à 256. Le paramètre pc désigne le pixel central où le filtre est appliqué. La taille du filtre BSIF à l'échelle S est réglé sur $d \times d$ où $d = 2 \times s + 1$.

I.3 HOG (Histogram of oriented gradients)

Un histogramme de gradient orienté (HOG) est une caractéristique utilisée en vision par ordinateur pour la détection d'objet. La technique calcule des histogrammes locaux de l'orientation du gradient sur une grille dense, c'est-à-dire sur des zones régulièrement réparties en fenêtre de détection, ou de la région d'intérêt) sur l'image [21]. Les principales étapes définies dans la mise en œuvre de l'algorithme de descripteur HOG sont comme suit:

1. Calcul du gradient

La première étape de la méthode est le calcul du gradient, la méthode la plus courante pour cela consiste à appliquer un filtre dérivatif 1-D centré, dans les directions horizontales et verticales. Les masques $[1,-0,1]$ et $[-1,0,1]^T$ sont utilisés pour cela: Dans le cas des images couleurs, le gradient est calculé séparément pour chaque composante, et on retient pour chaque pixel le gradient de plus grande norme. D'autres types de masques plus complexes ont été testés, comme des filtres de Sobel 3×3 , ou des masques diagonaux, ou non centrés.

2. Construction de l'histogramme

La seconde étape est la création des histogrammes de l'orientation des gradients. Ceci est fait dans des cellules carrées de petite taille (de 4×4 à 12×12 pixels). Chaque pixel de la cellule vote alors pour une classe de l'histogramme, en fonction de l'orientation du gradient à ce point. Le vote du pixel est pondéré par l'intensité du gradient en ce point. Les histogrammes sont uniformes de 0 à 180° (cas non signé) ou de 0 à 360° (cas signé).

3. Formation des blocs

Une étape importante est la normalisation des descripteurs afin d'éviter les disparités dues aux variations d'illumination. Cette étape introduit également de la redondance dans le descripteur. Pour cela, les auteurs regroupent plusieurs cellules dans un bloc, qui est l'unité sur laquelle est effectuée la normalisation. Les blocs se recouvrent, donc une même cellule participe plusieurs fois au descripteur final, comme membre de blocs différents.

Les descripteurs Locaux

Pour trouver les points-clés de l'échelle de l'emplacement, Lowe [11] utilise l'échelle de l'emplacement d'une image et fait la différence de la fonction Gaussien convoluté avec l'image en multipliant le premier σ par un facteur k .

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (2.12)$$

2. Localisation précise de points clés

Dans cette étape, l'objectif est de détecter des maxima et minima de différence de gaussienne dans l'espace d'échelle. Chaque point est comparé à ses 8 voisins dans l'image courante et 9 voisins chacun dans les échelles ci-dessus et ci-dessous. Une fois qu'un candidat point-clé est trouvé, un ajustement détaillé des données à proximité est effectué pour déterminer l'emplacement, échelle, et le rapport des courbures principales. Les courbures principales sont calculées à partir d'une fonction Hessian.

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (2.13)$$

Les valeurs propres de H sont proportionnelles aux courbures principales, de sorte que deux valeurs propres ne doit pas trop différentes. La matrice Hessienne a été utilisée pour éliminer les points de bord.

3. Affectation d'orientation

Attribuer une orientation à chaque point clé basée sur les caractéristiques locales de l'image. Un histogramme est formé en quantifiant les orientations dans 36 bins. Chaque bin dans l'histogramme est correspondre à une orientation de l'image. Pour la même échelle et l'emplacement, il pourrait y avoir plusieurs points-clés avec des orientations différentes.

4. Calcul de descripteurs de points-clés

Au ce niveau, chaque point clé a un emplacement, échelle et une orientation pour calculer un descripteur de la zone d'image locale de chaque point-clé qui est distinctif et invariant que possible aux variations telles que des changements de point de vue et d'éclairage.

Les descripteurs Locaux

Le descripteur SIFT prend un temps long pour décrire une image, pour cela une autre extension a été apparue c'est le descripteur SURF (Speeded Up Robust Features). Ce dernier est très rapide en le comparant à SIFT.

I.4.1 SURF (Speeded Up Robust Features)

C'est une extension de SIFT, il détecte des points d'intérêts invariants aux changements d'échelle et aux rotations. Afin de réduire le temps de traitement et accélérer la vitesse, SURF utilise les images intégrales pour le calcul de la convolution. L'image intégrale est une représentation intermédiaire pour l'image et contient la somme des valeurs de pixel au niveau de gris.

I.5 DAISY

Le descripteur DAISY proposé par Tola et al. en 2008 [55] est généré en se basant sur une approche locale en s'inspirant des avantages de la méthode SIFT. DAISY a pour objectif d'accélérer le temps de calculs et d'améliorer la gestion des invariances. L'idée principale est de remplacer les calculs de gradients de la méthode SIFT par des filtres de dérivées gaussiennes orientées. Fortement nuancées par la méthode du SIFT (descripteur suivant huit classes), les auteurs proposent de créer huit orientations définies par :

$$M_{\theta}^{\sigma} = g_{\sigma} * \left(\max \frac{aI}{\alpha\theta}, 0 \right) \quad (2.14)$$

Où θ est l'orientation de la dérivée. Chaque M_{θ} ainsi créée correspond à l'ensemble des gradients d'une orientation donnée, ayant une norme positive. Pour leur descripteur, les auteurs présentent un masque d'analyse circulaire d'origine décrit dans la figure II.4 constituée de 25 cercles définis suivant 3 échelles.

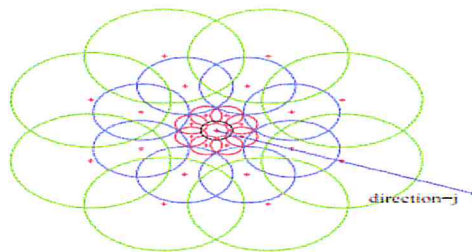


Figure II. 4: Masque d'analyse du descripteur Daisy, constituée de vingt-cinq cercles recalés suivant l'orientation locale du point d'intérêt [55].

Les descripteurs Locaux

Dés lors, un histogramme de gradients orientés est calculé à l'intérieur de chaque cercle. En se basant sur les orientations décrites par l'équation (2.15). L'histogramme h_θ en un point x est défini par :

$$h_\sigma = [M_1^\sigma(x), \dots, M_g^\sigma(x)]^T \quad (2.15)$$

Ce dernier est normalisé afin d'accroître l'invariance aux changements de luminosité et se note $h_\sigma(x)$. Le descripteur final, noté $D(x_0)$ se compose donc de vingt-cinq histogrammes (un par cercle) possédant chacun huit orientations et il est défini par :

$$D(x_0) = [h_{\sigma_1}^T(x_0), h_{\sigma_1}^T(l_1(x_0, R_1)), \dots, h_{\sigma_1}^T(l_N(x_0, R_1)), \\ h_{\sigma_1}^T(l_1(x_0, R_2)), \dots, h_{\sigma_1}^T(l_N(x_0, R_2)), \\ h_{\sigma_1}^T(l_1(x_0, R_3)), \dots, h_{\sigma_1}^T(l_N(x_0, R_3))]^T \quad (2.16)$$

où $(l_j(x_0, R_i))$, représente l'indice du cercle avoisinant. Les cercles sont ordonnés suivant trois rayons : R1, R2 et R3 et suivant huit orientations : 1 à N avec l'indice 1 pour l'orientation locale du point d'intérêt et $N = 8$.

I.6 LBP (local binary pattern)

L'idée du descripteur LBP est proposée par Ojala et al dans [56]. Son but est de décrire et de classifier les textures locales. Ce type de descripteur a été utilisé dans divers systèmes, comme les systèmes de reconnaissance de visage [57], des systèmes de détection de visage [44]...etc. La philosophie derrière ce descripteur est sa simplicité.

I.6.1 Principe

L'idée de cet opérateur de texture est d'assigner à chaque pixel un code dépendant des niveaux de gris de son voisinage. Le niveau de gris du pixel central (g_c) de coordonnées (x_c, y_c) est comparé à celui de ses P voisins (g_n) suivant l'équation (2.17). Tous les P voisins prennent une valeur 1 si leur valeur est supérieure ou égale au pixel du central et 0 sinon.

Les descripteurs Locaux

$$\text{sgn}(g_n - g_c) = \begin{cases} 1 & \text{si } (g_n - g_c) \geq 0 \\ 0 & \text{sinon} \end{cases} \quad (2.17)$$

A partir la fonction sgn , le code LBP du pixel courant est alors produit en concaténant ces valeurs pour former un code binaire, la valeur LBP est obtenue selon l'équation suivante:

$$\text{LBP}(X_c, y_c) = \sum_{n=0}^{p-1} 2^n \text{sgn}(g_n - g_c) \quad (2.18)$$

Où : (x_c, y_c) sont les coordonnées du pixel central.

La figure II.5 illustre un exemple de calcul par la méthode LBP de base.

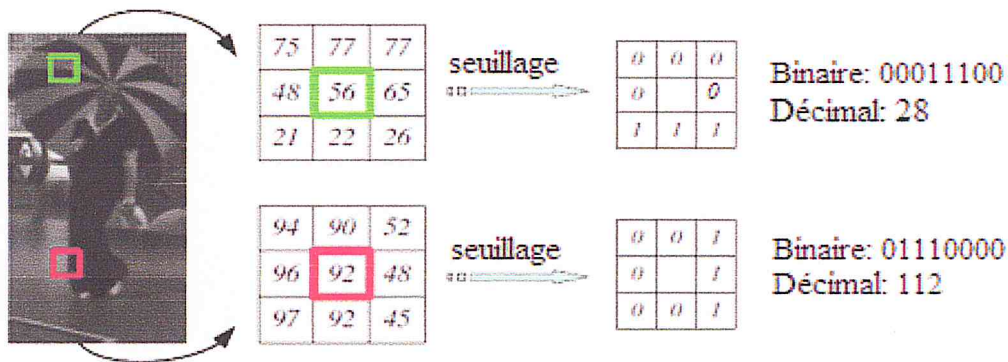


Figure II. 5: Exemple de calcul par la méthode LBP de base [55].

LBP a été étendu ultérieurement en utilisant des voisinages de différentes tailles. Dans ce cas, un cercle de rayon R autour du pixel central est considéré. Pour obtenir les valeurs des P points échantillonnés dans le voisinage pour tout rayon R , une interpolation est nécessaire. On adopte la notation (P, R) pour définir le voisinage de P points de rayon R d'un pixel. La figure II.6 illustre trois voisinages pour des valeurs différentes de R et P . LBP possède un vecteur caractéristique de taille 2^P [58].

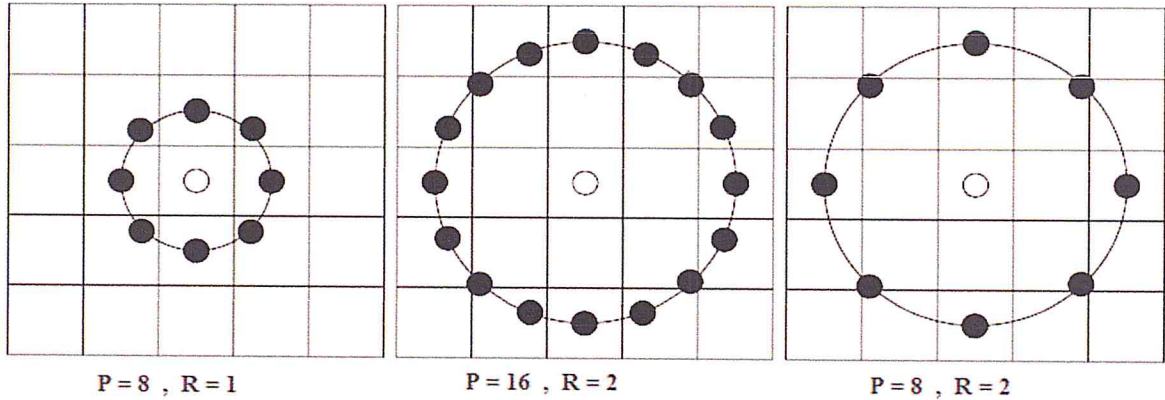


Figure II. 6: Trois voisinages pour des R et P différents [58].

La figure II.7 représente un exemple d'une image et histogramme LBP [58].

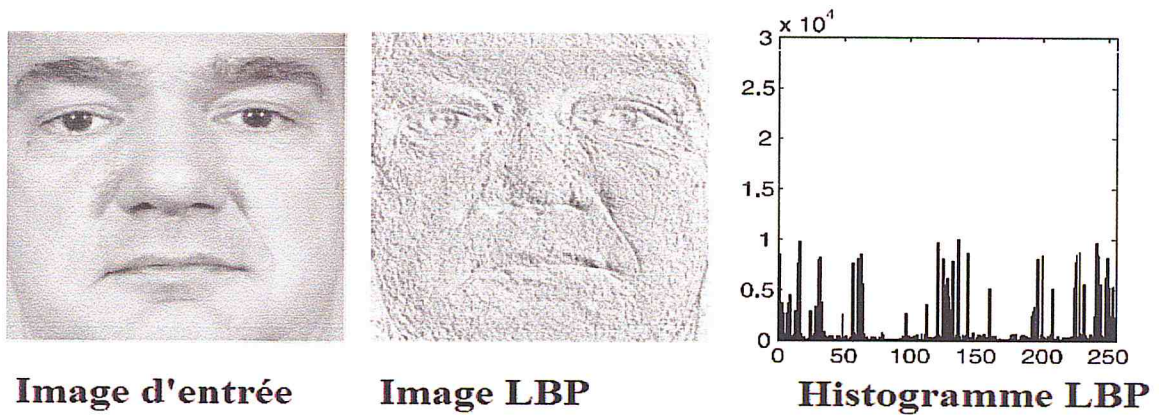


Figure II. 7: Exemple d'une image et histogramme LBP [58].

I.6.2 LBP Uniforme (Uniform Patterns)

LBP Uniforme [59] utilise une mesure de l'uniformité U , qui est le nombre de transitions binaires de 0 à 1 ou vice versa. On dit que l'opérateur LBP est uniforme si la mesure d'uniformité est au plus 2 transitions.

Par exemple, les motifs 00000000 (0 transition), 01110000 (2 transitions) 11001111 (2 transitions) sont uniformes alors que les motifs 11001001 (4 transitions) et 01010011 (6 transitions) ne sont pas uniforme.

LBP Uniforme possède cinq types de base de forme uniforme qui sont représentées dans la figure suivante II.8.

Les descripteurs Locaux

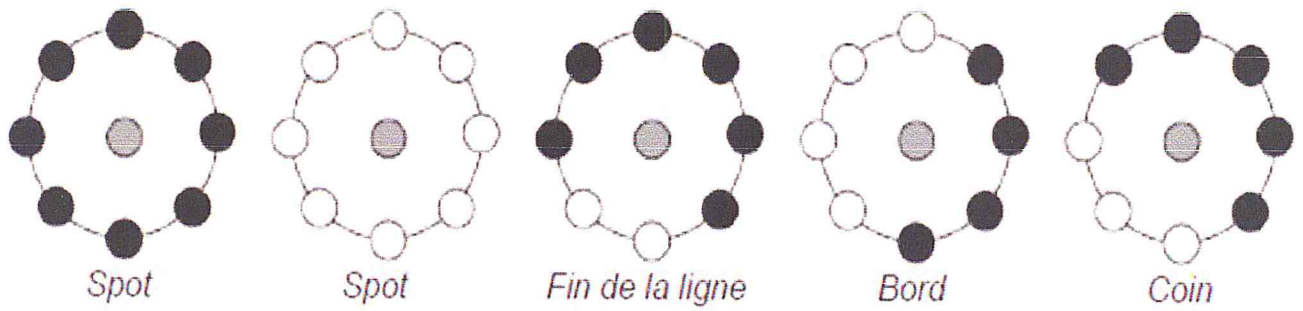


Figure II. 8: Textures particulières détectées par LBP [58].

Dans LBP uniforme il y a un bin de sortie séparée pour chaque forme uniforme, par contre les formes non uniformes sont affectées dans un seul bin. Le nombre des bins de sortie (vecteur caractéristique) de cet descripteur pour P voisins est $P(P-1)+3$.

Par exemple, il y a 59 bins de sortie pour les voisinages de $P=8$, et 243 bins pour les voisinages de $P=16$. La figure II.9 représente les 58 formes uniformes pour un voisinage de $(8, R)$.

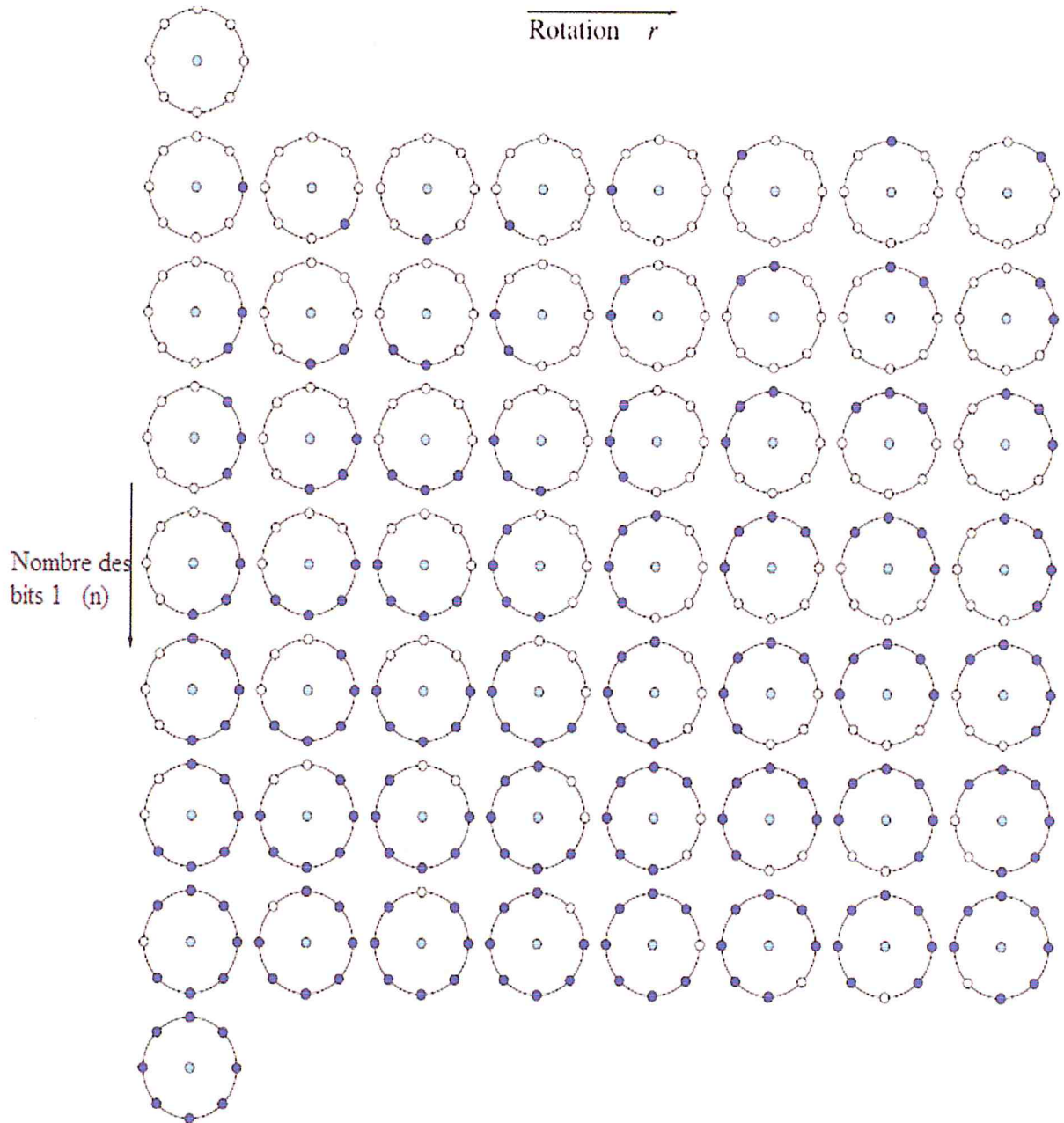


Figure II. 9: Les 58 formes uniformes pour un voisinage de $(8, R)$ [58].

Dans des nombreuses applications d'analyse de texture, il est souhaitable d'utiliser des caractéristiques qui sont invariantes ou robustes aux rotations de l'image d'entrée. Ce qui nécessite d'éliminer les rotations par nouvel opérateur LBP de rotation invariant.

I.6.3 LBP de rotation invariant

La rotation de l'image d'entrée comporte deux effets: chaque voisinage local est mis en rotation dans un autre emplacement de pixel, et dans chaque voisinage, les points

Les descripteurs Locaux

d'échantillonnage sur le cercle entourant le point central sont mis en rotation dans une orientation différente.

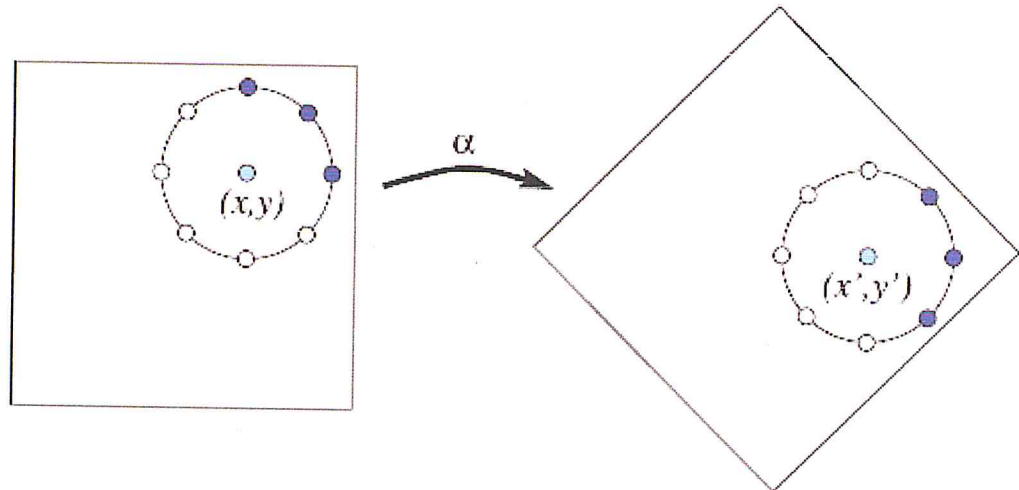


Figure II. 10: Effet de rotation de l'image sur des points dans les voisinages de circulaires [58].

Dans cette application, chaque code binaire LBP est circulairement tourné dans sa valeur minimale [58].

$$LBP_{P,R}^{ri} = \min_i \text{ROR} (LBP_{P,R}, i) \quad (2.19)$$

Où $\text{ROR} (x, i)$ effectue un décalage (rotation) à droite au niveau des bits i fois. En termes de pixels d'image, la formule (2.19) précédente correspond simplement à tourner les pixels voisins dans le sens des aiguilles d'une montre autant de fois que la valeur minimum de bits les plus significatifs.

Par exemple, pour les codes LBP $(1000010)_2$, $(00101000)_2$ et $(00000101)_2$, après la rotation appliquée par l'équation (2.12), le code de rotation invariant doit être le code minimal $(00000101)_2$.

Notons que les descripteurs cités précédemment sont utilisables pour décrire les caractéristiques spatiales (la texture statique) qui sont appliquées seulement sur les images, tandis que, dans le cas d'une vidéo la robustesse de ces descripteurs est diminué (perte d'information temporelle), ce qui est nécessaire d'introduire des descripteurs spatio-temporel.

Les descripteurs Locaux

Si les coordonnées de $g_{t,c}$ sont (x_c, y_c, t_c) , les coordonnées de $g_{t,p}$ et ceux de $g_{t,c \pm L,p}$ sont calculées de la manière suivante :

$$((x_c - R \sin(2 \pi p / P), (y_c + R \cos(2 \pi p / P)), t_c). \quad (2.21)$$

et les coordonnées de $g_{t,c \pm L,p}$ sont :

$$((x_c - R \sin(2 \pi p / P), (y_c + R \cos(2 \pi p / P)), t_c \pm L). \quad (2.22)$$

La valeur du *VLBP* dans un voisinage local V est obtenue en soustrayant le niveau de gris du pixel du centre ($g_{t,c}$) aux niveaux de gris des pixels ($g_{t,p}$) appartenant à la région symétrique circulaire, la distribution devient [61]:

$$V \approx v(g_{t,c-L,c} - g_{t,c}, g_{t,c-L,0} - g_{t,c}, \dots, g_{t,c-L,P-1} - g_{t,c}, g_{t,c,0} - g_{t,c}, \dots, g_{t,c,P-1} - g_{t,c}, g_{t,c+L,0} - g_{t,c}, \dots, g_{t,c+L,P-1} - g_{t,c}, g_{t,c+L,c} - g_{t,c}). \quad (2.23)$$

Pour obtenir l'invariant par rapport à la mise à l'échelle de niveau de gris, seuls les signes des différences sont pris en considération au lieu des valeurs exactes de chaque pixel [61]:

$$V \approx v(\text{sgn}(g_{t,c-L,c} - g_{t,c}), \text{sgn}(g_{t,c-L,0} - g_{t,c}), \dots, \text{sgn}(g_{t,c-L,P-1} - g_{t,c}), \text{sgn}(g_{t,c,0} - g_{t,c}), \dots, \text{sgn}(g_{t,c,P-1} - g_{t,c}), \text{sgn}(g_{t,c+L,0} - g_{t,c}), \dots, \text{sgn}(g_{t,c+L,P-1} - g_{t,c}), \text{sgn}(g_{t,c+L,c} - g_{t,c})). \quad (2.24)$$

Où le paramètre sgn est la fonction de seuillage définie dans l'équation (2.1) .

Pour simplifier l'expression de V , l'utilisation de $V = v(v_0, \dots, v_q, \dots, v_{3P+1})$, et q correspond à l'index des valeurs dans V . Un coefficient binomial 2^q est attribué pour chaque signe $\text{sgn}(g_{t,p} - g_{t,c})$, après transformation de V en un numéro unique $VLBP_{L,P,R}$ qui caractérise la structure spatiale de la texture dynamique du volume local [61] :

$$VLBP_{L,P,R} = \sum_{q=0}^{3P+1} v_q 2^q \quad (2.25)$$

Les descripteurs Locaux

$$\begin{aligned}
 V2 \approx v([sgn(g_{tc-L,c} - g_{tc,c}), & [sgn(g_{tc-L,0} - g_{tc,c}), \dots, sgn(g_{tc-L,P-1} - g_{tc,c})], \\
 [sgn(g_{tc,0} - g_{tc,c}), \dots, & sgn(g_{tc,P-1} - g_{tc,c})], [sgn(g_{tc+L,0} - g_{tc,c}), \dots, sgn(g_{tc+L,P-1} - g_{tc,c})], \\
 [sgn(g_{tc+L,c} - g_{tc,c})]) &
 \end{aligned} \tag{2.26}$$

La division organisée en ordre en ordre et V_{preN} , V_{curN} et V_{posN} représente le code LBP de points voisins dans l'image précédente, celle de l'image du centre et l'image suivante, respectivement. Tandis que V_{preC} et V_{posC} représentent les valeurs binaires des pixels centraux dans l'image précédente et suivante [61].

$$LBP_{t,P,R} = \sum_{p=0}^{P-1} sgn(g_{t,p} - g_{t,c}) 2^p \tag{2.27}$$

Où $t=t_c - L, t_c, t_c + L$

La formule précédente permet d'obtenir le code LBP $_{tc-L,P,R}$ (pour l'image précédente), LBP $_{tc,P,R}$ (pour l'image centrale) et LBP $_{tc+L,P,R}$ (pour l'image suivante). Et l'utilisation en premier lieu de la formule ci-dessous pour enlever l'effet de la rotation [61]:

$$\begin{aligned}
 VLBP_{L,P,R}^i &= \min\{(VLBP_{L,P,R} \text{ and } 2^{3P+1}) \\
 &+ \text{ROL}(\text{ROR}(LBP_{tc+L,P,R}, i), 2P+1) \\
 &+ \text{ROL}(\text{ROR}(LBP_{tc,P,R}, i), P+1) \\
 &+ \text{ROL}(\text{ROR}(LBP_{tc-L,P,R}, i), 1) \\
 &+ (VLBP_{L,P,R} \text{ and } 1) \quad i = 0, 1, \dots, P-1\}
 \end{aligned} \tag{2.28}$$

Où la fonction $ROR(x, i)$ effectue un décalage de i fois à droite du bit circulaire sur le P bits, et $ROL(y, j)$ effectue un décalage de j fois à gauche sur $(3P + 2)$ bits. En termes de pixels de l'image, la formule (2.28) correspond simplement à faire tourner les voisins dans

Les descripteurs Locaux

trois cadres séparés dans le sens horaire. Cela arrive de façon synchrone de sorte qu'une valeur minimale est sélectionnée comme code de la rotation invariante VLBP.

Par exemple, pour le code VLBP originale $(1,1010,1101,1100,1)_2$ ses codes après une rotation vers la gauche de 90, 180, 270 degrés sont respectivement $(1,0101,1110,0110,1)_2$, $(1,1010,0111,0011,1)_2$ et $(1,0101,1011,1001)_2$. Leur code de rotation invariant doit être $(1,0101,0111,0011,1)_2$.

I.7.3 LBP-TOP (LBP- Tree Orthogonal Plan)

Dans l'opérateur VLBP, le paramètre P détermine le nombre de fonctionnalités. Une grande valeur de P produit un long histogramme, tandis qu'une petite valeur de P rend le vecteur de caractéristiques plus court. Ceci signifie également perdre plus d'informations. Lorsque le nombre de points voisins augmente, le nombre de motifs de base VLBP devient très grand (2^{3P+2}). En raison de cette augmentation rapide, il est difficile d'étendre VLBP pour un grand nombre de points voisins, ce qui limite sa robustesse [62]. En même temps, lorsque l'intervalle de temps $L > 1$, les trames voisines avec une variance inférieure à L temps sera omise.

Pour résoudre ces problèmes, en proposant un descripteur simplifié par la concaténation des motifs binaires locaux (LBP) sur les trois plans orthogonaux XY , XT et YT en fonction de ses voisins dans ces trois directions, Les codes de LBP sont extraits depuis les plans XY , XT et YT , qui sont appelés $XY-LBP$, $XT-LBP$ et $YT-LBP$. Cette opération est répétée jusqu'à ce que tous les pixels de chaque trame sont pris en considération pour obtenir les histogrammes correspondants, qui sont concaténés dans un seul histogramme (figure II.13).

Donc la texture dynamique est codée par $XY-LBP$, $XT-LBP$ et $YT-LBP$, tandis que $XY-LBP$ représente des informations de domaine spatial, $XT-LBP$ et $YT-LBP$ représentent des informations de domaine spatio-temporel. Avec cette approche, La taille du vecteur caractéristique de LBP-TOP est seulement $(3*2^P)$, beaucoup plus petite que (2^{3P+2}) , ce qui rend l'extension à des nombreux points voisins plus facile et réduit également la complexité de calcul.

Les descripteurs Locaux

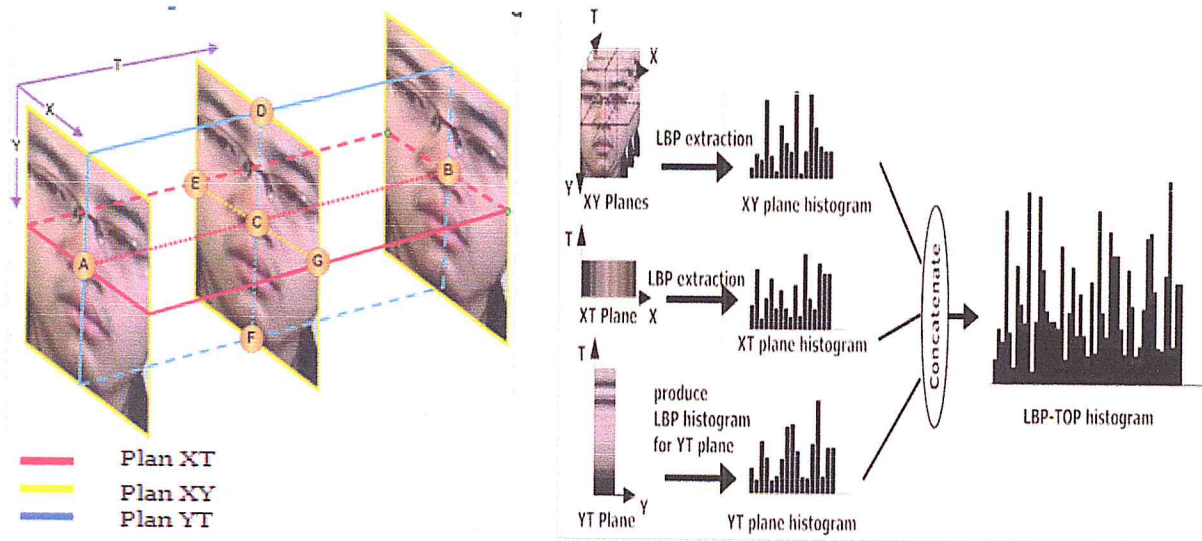


Figure II. 13: Principe de LBP-TOP [60].

I.7.4 La différence entre les descripteurs VLBP et LBP-TOP

En résumé, le tableau suivant récapitule la différence entre VLBP et LBP-TOP [58].

VLBP	LBP-TOP
<ul style="list-style-type: none"> • VLBP utilise trois images consécutives dont l'image centrale contient tous les pixels centraux. • VLBP considère tous les points voisins de trois cadres parallèles, ce qui tend à rendre le vecteur de caractéristiques trop grand. • La taille du vecteur caractéristiques est (2^{3P+2}) qui correspond à un grand vecteur caractéristiques.. 	<ul style="list-style-type: none"> • LBP-TOP utilise trois images consécutives également, mais sont elles sont réparties sur les plans orthogonaux qui se croisent dans le pixel central • LBP-TOP considère les caractéristiques des distributions de chaque plan distinct, puis les concatène, ce qui rend le vecteur caractéristique beaucoup plus court lorsque le nombre de points voisins augmente. • La taille de vecteur caractéristiques est $(3*2^P)$. Ce qui correspond à un vecteur court.

Tableau II. 1: la différence entre VLBP et LBP-TOP.

Conclusion

Dans ce chapitre, nous avons étudié de manière générale quelques descripteurs locaux généralement utilisés dans les systèmes de détection de visage afin de déterminer les points forts et les points faibles de chacun. Le descripteur LBP et ses variantes est présenté de manière détaillée car nous l'avons utilisé pour développer notre système de détection de copie vidéo qui fera l'objet du prochain chapitre.



Chapitre III

Résultats et Evaluation

Introduction

Le but du travail est de concevoir et de réaliser un système de détection de copie vidéo basé sur le contenu (CBVD) robuste au maximum d'attaques de traitement d'images. Après une étude des différents descripteurs locaux, notre choix s'est porté sur le descripteur LBP et ses variantes en raison de son pouvoir discriminant et sa simplicité de calcul. Le système de copie vidéo est développé selon le descripteur LBP et ses variantes

LBP Uniforme, LBP Rotationnel et LBP Uniforme Rotationnel et afin d'exploiter la caractéristique spatio-temporelle de la vidéo, nous avons utilisé les descripteurs spatio-temporel VLBP, VLBP Rotationnel et LBP-TOP. Une étude empirique entre ces derniers est effectuée en se basant sur deux paramètres de similarité MahCosine et χ^2 . Le système de détection de copie vidéo final est réalisé avec le descripteur LBP robuste au maximum d'attaques.

I. Conception du système de détection de copie vidéo

Le schéma synoptique du système de détection de copie vidéo réalisé est typique, il comprend trois principaux qui sont présentés dans la figure III.1:

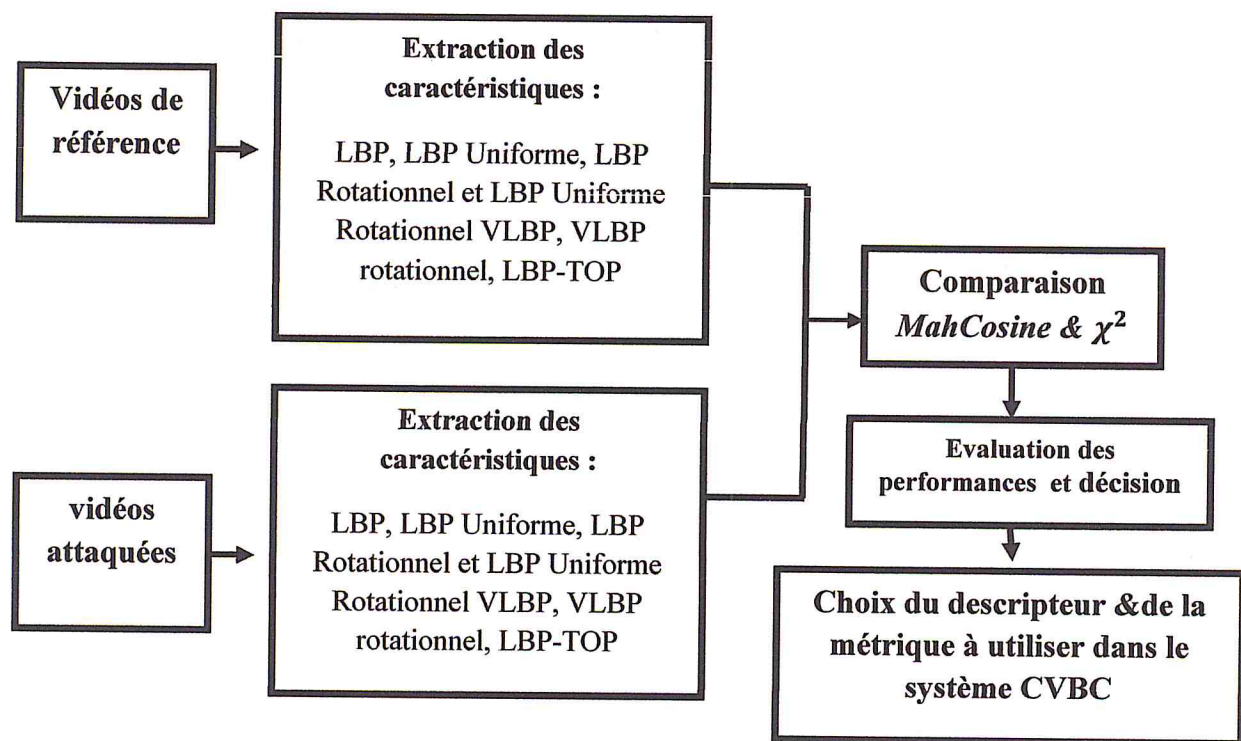




Figure III. 1: Conception du système de détection de copie vidéo.







I.1 Base de données de test

Les vidéos de tests utilisées sont issues de la base de données vidéo *TRECVID 2009* [64]. Cette dernière comprend des vidéos de type scientifiques, reportages, des documentaires, des programmes éducatifs, des événements sportifs, des émissions de télévision, des dessins animés et des vidéos d'archives. TRECVID est utilisée dans plusieurs thèmes de recherche tels que l'extraction des caractéristiques de haut niveau, la recherche (entièrement automatique, manuellement assistée, ou interactif), la détection de copie, ou la détection des événements de surveillance.

Vue le matériel utilisé, seulement 30 vidéos de durée de 9 s chacune sont utilisées pour construire la base de données vidéos de référence de taille 1024X576 et de format AVI. La vidéo cible ou requête, est issue d'une base de données comprenant 270 vidéos ayant subi des transformations géométriques et des manipulations de traitement d'images. Le tableau III.1 ci-dessus regroupe les différentes manipulations (attaques) effectuées.

Attaque	Description
Blur (Flou) 	L'image floue est obtenue par convolution de l'image avec une fonction de moyenne pondérée des pixels voisins. Cette transformation résulte une perte de détails sur l'image.
Cropping 	Le cropping consiste à définir des zones rectangulaires à l'intérieur d'un cadre d'image de la vidéo et supprimer les pixels de l'image qui sont en dehors de cette région. Cette transformation produit généralement des images noires ou des cadrages sur un ou plusieurs côtés du segment vidéo. La bordure noire autour de l'image sont les pixels qui ont été supprimés ou recadrés.
Caption	C'est l'insertion d'un motif ou d'un texte sur des segments de la vidéo. Cette transformation introduit de nouveaux bords et des régions d'intérêts, et provoque des changements sur l'information couleur.

Résultats et Evaluation

 <p>CDTA copyright VCD project</p>	
<p>Gamma</p> 	<p>Est une déformation de la luminosité. La correction gamma est un principe fondamental dans le flux de reproduction des images, il modifie la distribution totale des images.</p>
<p>Noise (Bruit)</p> 	<p>Le bruit est introduit d'une manière telle qu'il présente une distribution de fréquence.</p>
<p>Compression</p> 	<p>il consiste à réduire la quantité de données, en minimisant l'impact sur la qualité visuelle de la vidéo.</p>
<p>Hue</p> 	<p>Cette transformation permet de modifier la teinte</p>
<p>Flipping (Clip)</p> 	<p>C'est la mise en rotation autour d'un axe vertical dans le centre de l'image.</p>

Résultats et Evaluation

	V1	V2	V3	V4	V30
V1	C	NC	NC	NC	NC	NC
V2	NC	C	NC	NC	NC	NC
V3	NC	NC	C	NC	NC	NC
V4	NC	NC	NC	C	NC	NC
.....	NC	NC	NC	NC	C	NC
.....	NC	NC	NC	NC	NC	C
.....	NC	NC	NC	NC	NC	C
.....	NC	NC	NC	NC	NC	C
.....	NC	NC	NC	NC	NC	C
.....	NC	NC	NC	NC	NC	C
V30	NC	NC	NC	NC	NC	C

Figure III. 2: La matrice de comparaison.

La distance entre chaque vidéo d'origine et toutes les vidéos d'attaques est calculée afin de construire une matrice de comparaison qu'on appelle *score*. Les valeurs de la diagonale principale représentent les scores des vidéos copies (True Score, *TS*) tandis que les autres valeurs sont les scores des vidéos non copies (False Scores, *FS*). La figure III.2 représente une matrice de comparaison.

I.5 Le module de décision et performances du système réalisé

Ce module permet de vérifier l'identité affirmée par une vidéo ou déterminer l'identité d'une vidéo en se basant sur le degré de similitude entre les descripteurs des vidéos attaquées et ceux stockés dans la base de données. La similarité des descripteurs se base sur la mesure des performances du système CBVD réalisé. Des métriques standard de performance des systèmes de détection sont utilisées. Celles-ci sont énumérées comme suit [65]:

- **Taux de vérification (Verification Rate)** : Il mesure le pourcentage des entrées qui sont correctement vérifiées.
- **Taux de Faux Rejet (TFR) ou (False Reject Rate, FRR)** : Ce taux représente le pourcentage de vidéos censées être copies mais qui sont rejetées par le système.
- **Taux de Fausse Acceptation (TFA) ou (False Accept Rate, FAR)** : Ce taux représente le pourcentage de vidéos censées ne pas être copies mais qui sont tout de même acceptées par le système.

- **Taux d'Égale Erreur TEE ou (Equal Error Rate, EER)** : Ce taux est calculé à partir des deux premiers critères TFR et TFA et constitue un point de mesure de performance courant. Ce point correspond à l'endroit où TFR est égal à TFA, c'est-à-dire le meilleur compromis entre les faux rejets et les fausses acceptations.
- **Receiver Operating Characteristic (ROC)** : La courbe ROC est une métrique graphique visualisant compromis entre TFA et le TFR liées à un seuil variable. Cette courbe est utilisée généralement dans un système d'authentification.
- **Seuil de détection** : En général, les valeurs de TFA et TFR peuvent se recouvrir. Pour cela, il est nécessaire de fixer un seuil à partir duquel, nous décidons que les vidéos sont acceptées ou rejetées, comme représenté dans la figure III.3.

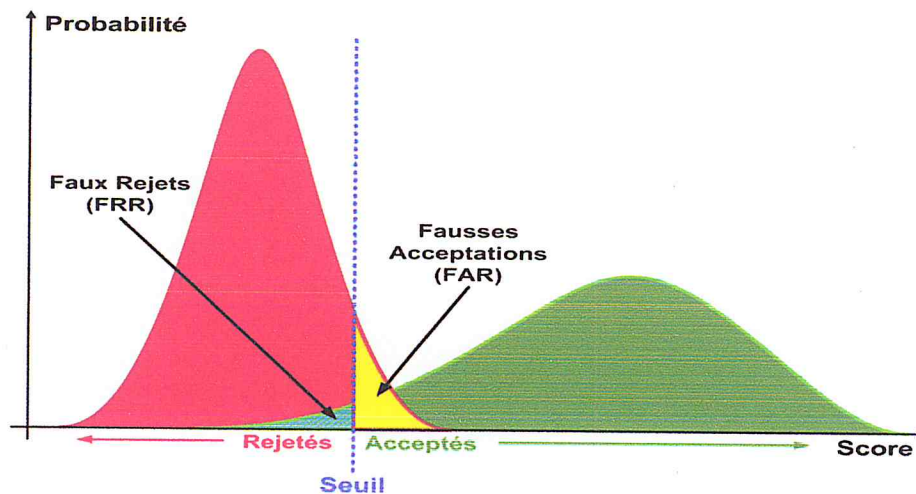


Figure III. 3: Illustration du TFA et TFR [65].

Les tests et les résultats obtenus dépendent du domaine d'application des descripteurs. Dans la première partie de ce paragraphe, nous présenterons les résultats résultants de l'application des descripteurs spatiaux et dans la deuxième partie, figureront les résultats obtenus par l'application des descripteurs spatio-temporels.

II. Résultats expérimentaux par l'application des descripteurs LBP et ses variantes dans le domaine spatial.

Différents descripteurs de LBP opérant dans le domaine spatial sont testés. L'utilisation du terme domaine spatial signifie que l'application des descripteurs s'effectue sur chaque image des séquences en entrée. Dans ce domaine, les caractéristiques des deux bases, la base de vidéos de référence et celle attaquée sont extraites par l'application des descripteurs LBP de base, LBP Uniforme, LBP Rotationnel et LBP Uniforme Rotationnel. Les tests sont effectués

Résultats et Evaluation

pour différents nombre de voisins et pour différents rayons. Les descripteurs obtenus sont sauvegarder dans des matrices, sous format des fichiers *.mat : une matrice pour des vidéos de référence et neuf matrices pour les vidéos attaquées. A partir de ces matrices 10 histogrammes sont construits : un histogramme représentant les caractéristiques des vidéos de référence et les neuf histogrammes représentent les caractéristiques des vidéos attaquées. La figure III.4 illustre les histogrammes obtenus par l'application du descripteur LBP avec un voisinage de P égale à 8.

Résultats et Evaluation

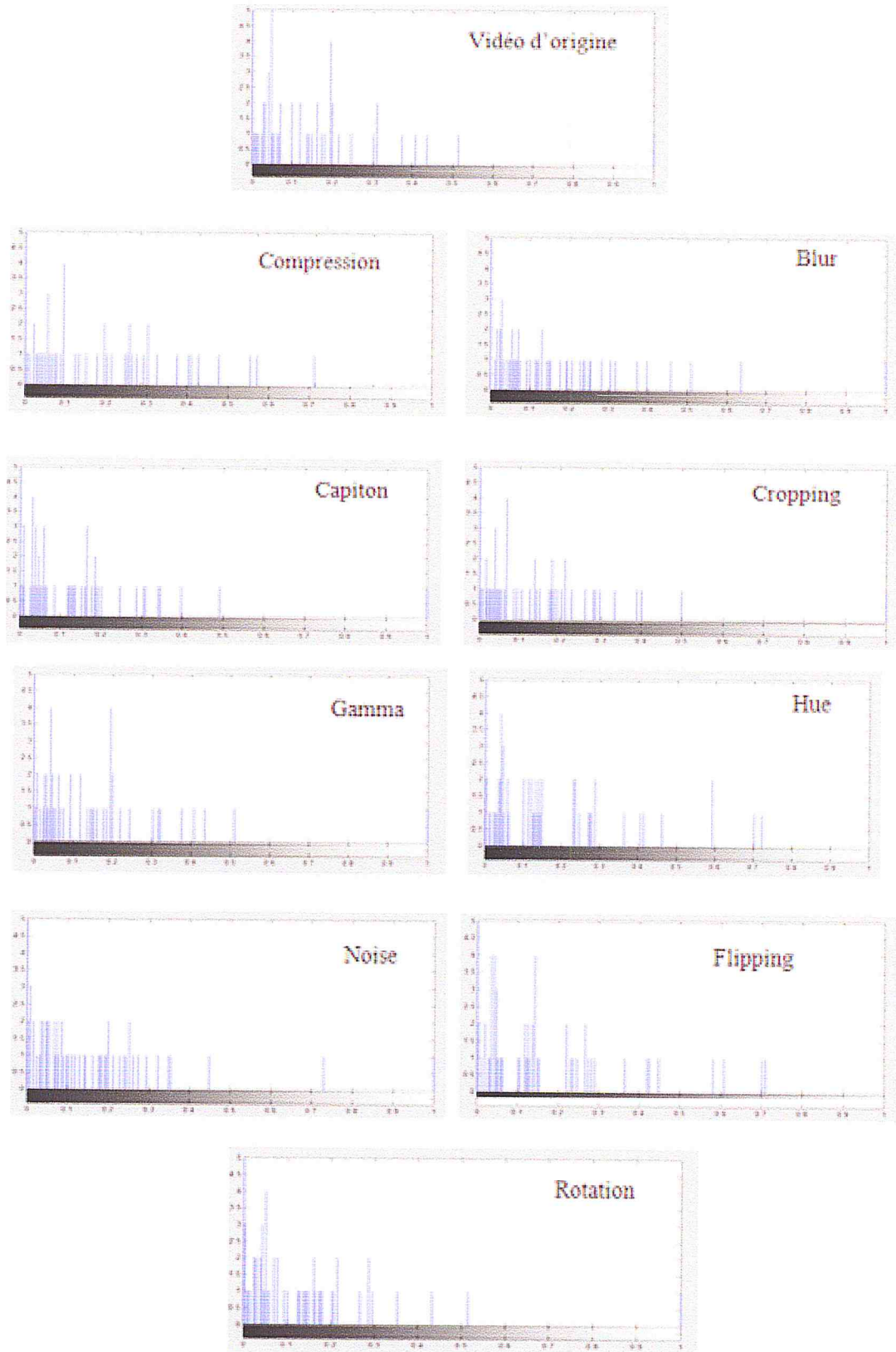


Figure III. 4: les histogrammes de LBP avec un voisinage de P égale à 8.

Résultats et Evaluation

Nous avons constaté que certaines vidéos attaquées n'ont pas la même taille que celle des vidéos de référence, et ceci rend la comparaison impossible. Pour résoudre ce problème, dans notre conception, nous avons incorporé l'algorithme d'Analyse en Composantes Principales (PCA), algorithme efficace et couramment utilisé en réduction de dimension.

L'algorithme PCA est une méthode globale utilisant en premier lieu les niveaux de gris des pixels d'une image. Le principe selon lequel on peut construire un sous-espace vectoriel en ne retenant que les « meilleurs » vecteurs propres, tout en conservant beaucoup d'information utile, fait du PCA un algorithme efficace et couramment utilisé en réduction de dimensionnalité. Les étapes de PCA sont les détaillés dans l'annexe 1.

La figure III.5 résume les étapes d'extraction de caractéristiques dans l'axe spatial.

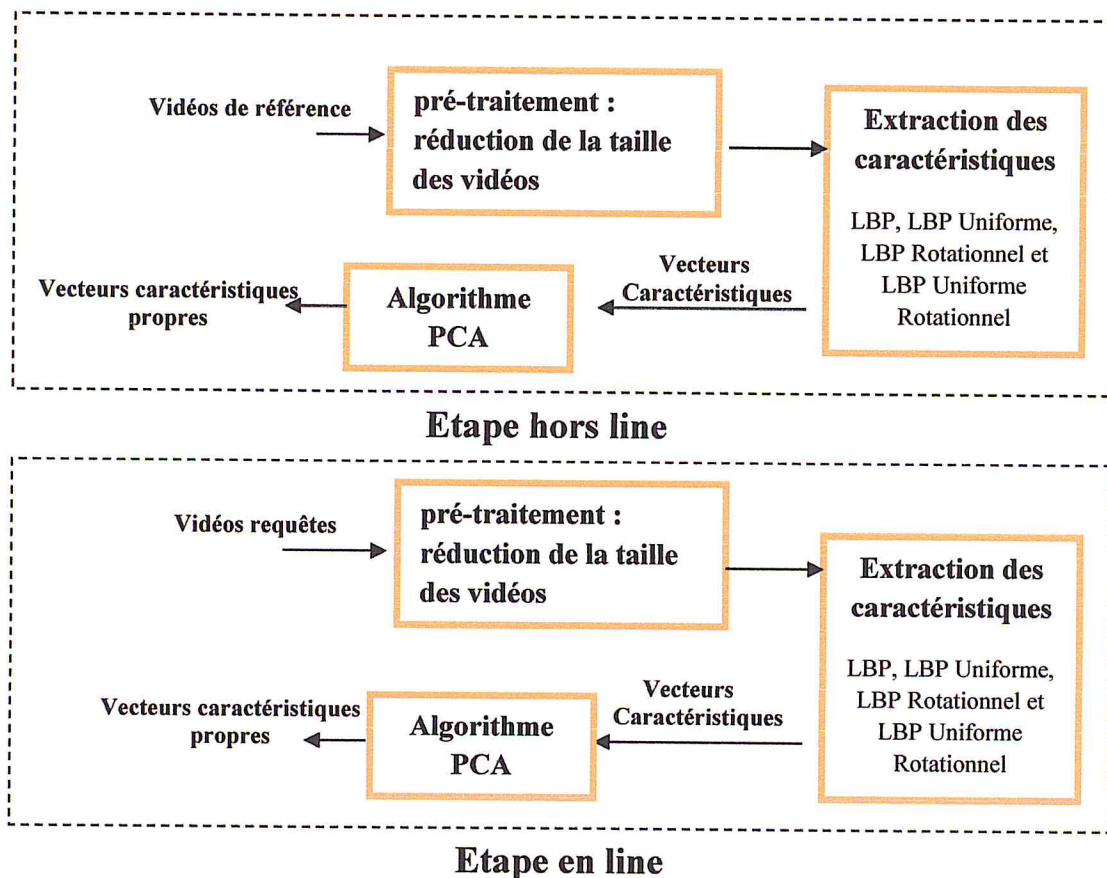


Figure III. 5: Extraction des caractéristiques dans l'axe spatial.

II.1 Performances du système dans le domaine spatial

La performance de chaque descripteur selon les différentes transformations est représentée par son taux de vérification TV . Ce dernier est exprimé par le taux inverse de faux rejet ($1 - TFR$) par rapport au taux de fausse acceptation TFA .

Ces mesures sont définies comme suit : soit N_T est le nombre total de tests de comparaison, FR le nombre de faux rejet et FA le nombre de fausse acceptation. sont définis comme suit, nous avons :

$$TFA = \frac{FA}{N_T}, \quad TFR = \frac{FR}{N_T}, \quad TV = 1 - TFR \quad (3.1)$$

II.1.1 Les résultats du descripteur LBP de base

Les résultats obtenus par l'application du descripteur LBP de base pour un voisinage p de 4, 8 et 16 en utilisant les deux métriques MahCosine et χ^2 sont regroupés dans le tableau III.2. Les figures illustrent les performances du descripteur LBP de base avec les différentes métriques utilisées :

Résultats et Evaluation

Les métriques	MahCosine			χ^2		
	P = 4	P = 8	P = 16	P = 4	P = 8	P = 16
Voisinage						
Attaques						
Blur	76.67	90	86.67	73.33	83.33	86.67
Crop	93.33	96.67	96.67	93.33	90	86.67
Caption	90	96.67	96.67	93.33	93.33	83.33
Gamma	56.67	56.67	53.33	56.67	70	73.33
Noise	83.33	83.33	86.67	93.33	83.33	86.67
Compression	73.33	80	90	60	70	76.67
Hue	100	100	100	100	100	86.67
Flipping	63.33	90	90	73.33	86.67	66.67
Rotation	63.33	53.33	56.67	76.67	56.67	56.67

Tableau III. 2: Les résultats de LBP base.

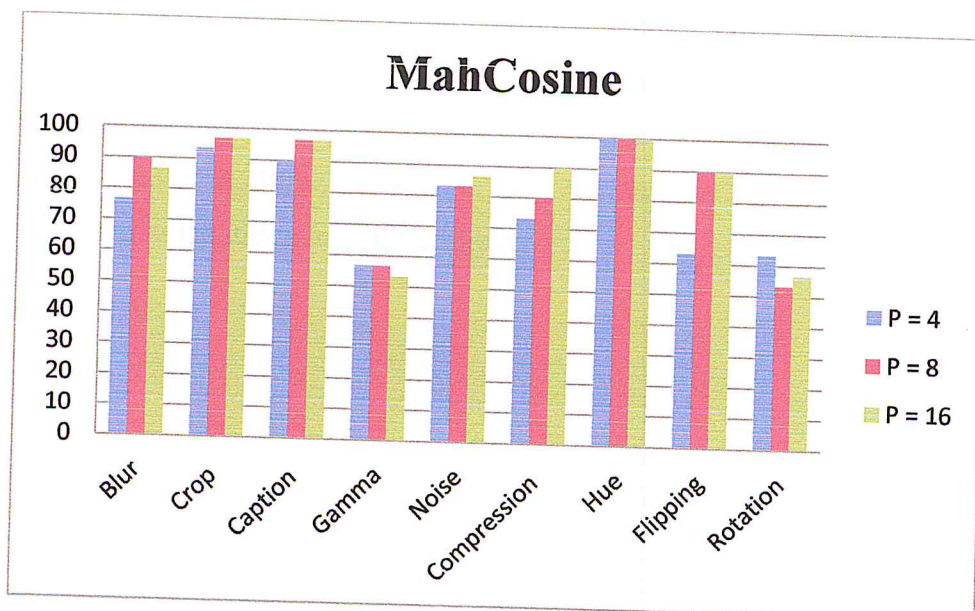


Figure III. 6: Performances du descripteur LBP de base selon la métrique MahCosine.

Résultats et Evaluation

Les métriques	MahCosine			χ^2		
	P = 4	P = 8	P = 16	P = 4	P = 8	P = 16
Voisinage						
Attaques						
Blur	73,33	83,33	73,33	76,67	63,33	60
Crop	93,33	96,67	80	3,33	30	56,67
Caption	86,67	93,33	80	23,33	76,67	3,33
Gamma	36,67	53,33	46,67	36,67	56,67	50
Noise	83,33	76,67	66,67	76,67	46,67	30
Compression	73,33	66,67	50	33,33	40	46,67
Hue	100	96,67	93,33	6,67	96,67	3,33
Flipping	76,67	86,67	86,67	3,33	86,6667	63,33
Rotation	63,33	56,67	56,67	36,67	50	46,67

Tableau III. 3: Les résultats obtenus avec le descripteur LBP uniforme.

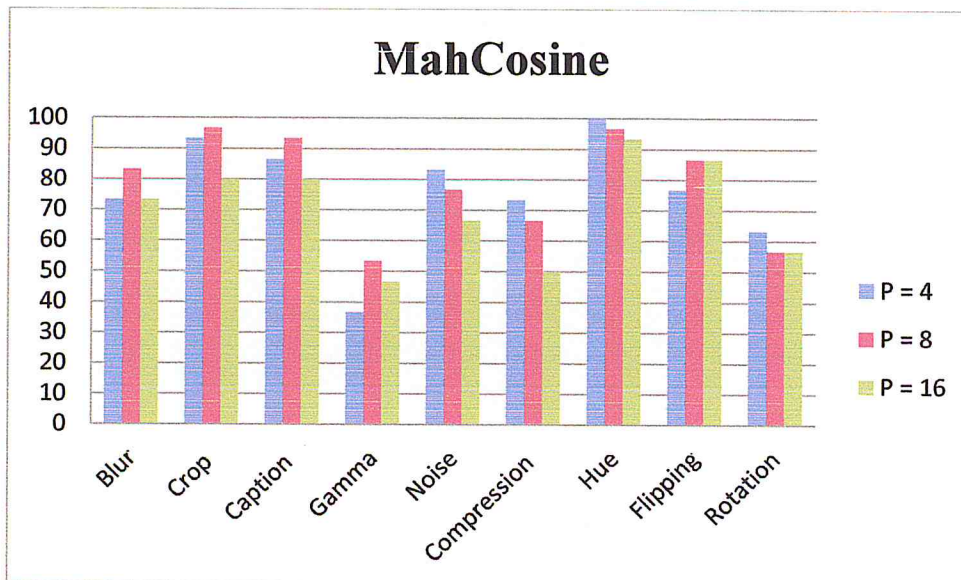


Figure III. 9: Les résultats de LBP uniforme avec MahCosine.

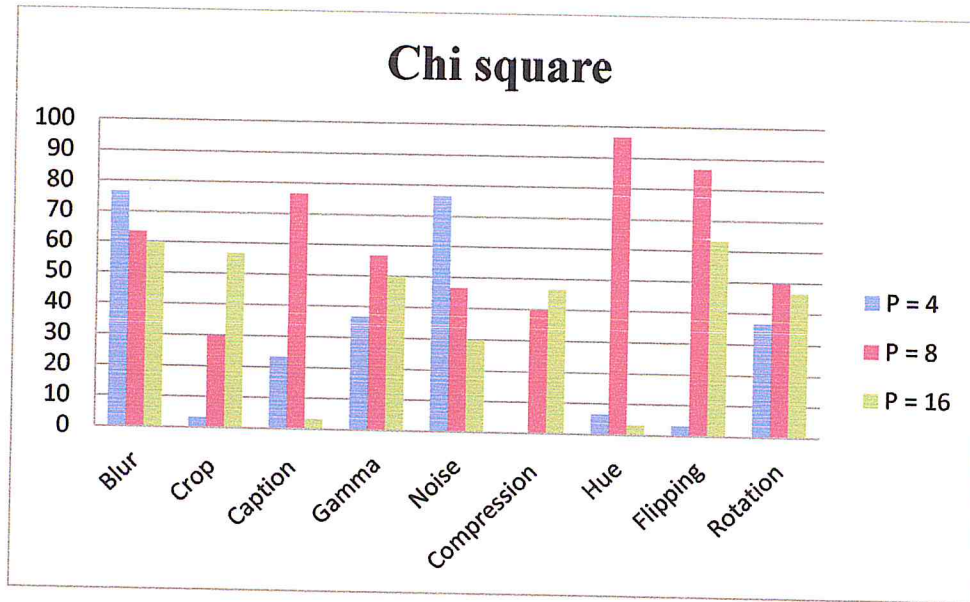


Figure III. 10: Les résultats de LBP uniforme avec χ^2 .

D'après les résultats illustrés sur le tableau III.3 et les figures III.9 et III.10, nous déduisons que le descripteur LBP uniforme pour $P=8$ donne de meilleurs résultats en terme de robustesses par rapport aux attaques de flou, « Cropping », « Caption », « Gamma » et le « Flipping » en utilisant la métrique MahCosine. Par contre pour les attaques bruit, changement de teinte, compression et rotation le descripteur LBP uniforme avec un voisinage de 4 donne des meilleurs résultats. Nous pouvons conclure que LBP uniforme a de meilleures performances avec la métrique MahCosine en utilisant le voisinage $P=8$.

La figure III.11 illustre les courbes ROC des différentes attaques dans l'ensemble des tests effectués en utilisant LBP uniforme pour un voisinage égal à 8 et la métrique MahCosine.

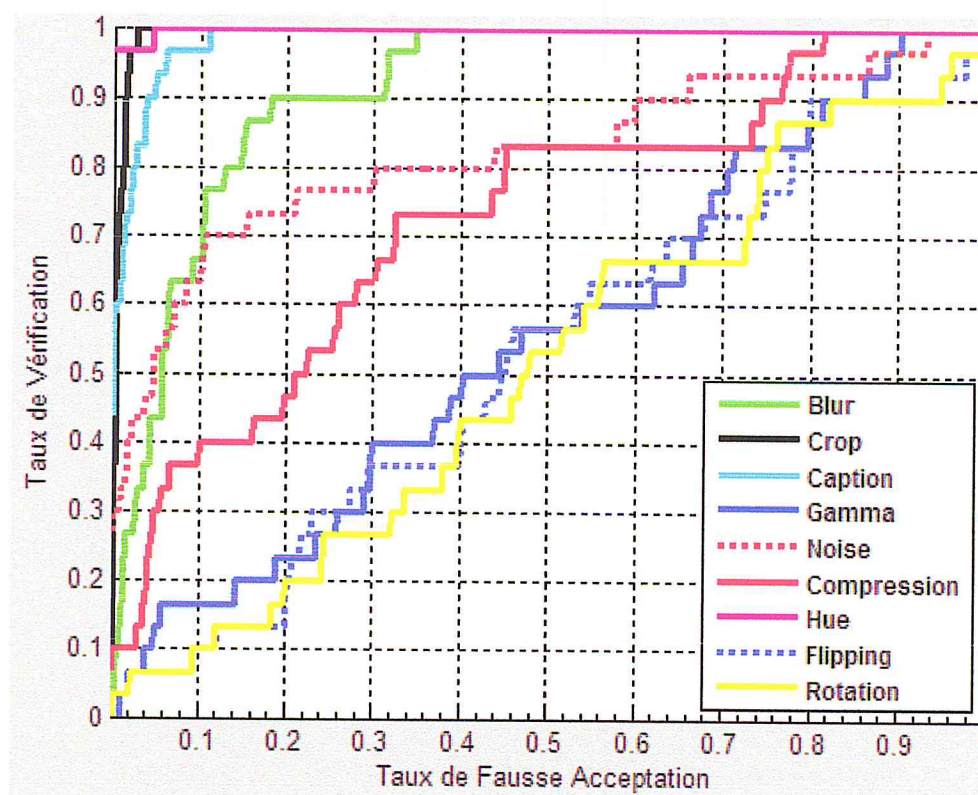


Figure III. 11: le ROC de différentes attaques avec LBP Uniforme, MahCosine et $P=8$.

D'après la figure III.11, nous constatons que LBP uniforme avec un voisinage de $P=8$ est très robuste à la modification de teinte (Hue) et aux transformations de « Cropping » et « Caption ». Par contre, il est moins performant dans les trois transformations « Gamma », « Flipping » et rotation.

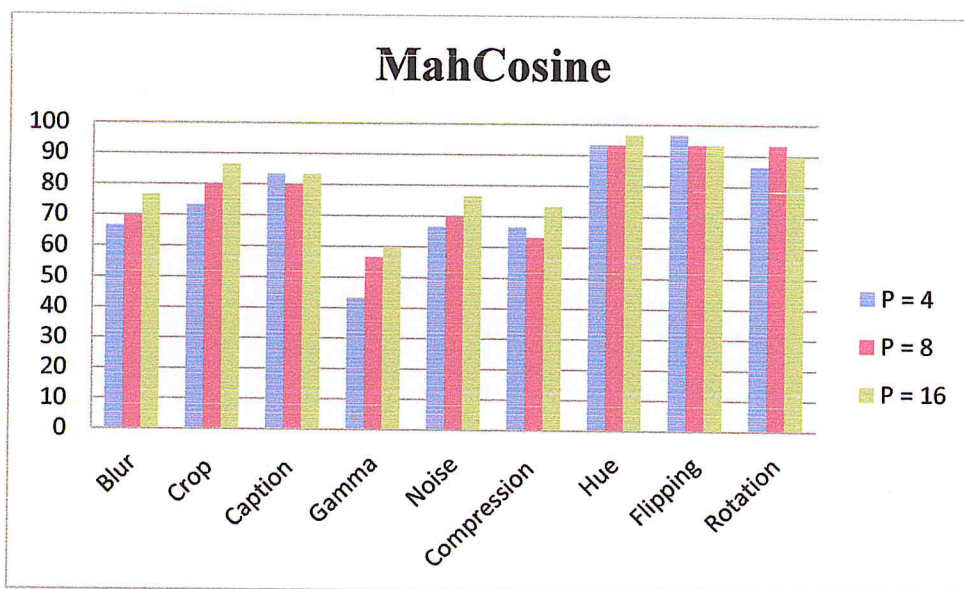
II.1.3 Les résultats du descripteur LBP rotationnel

Des tests avec le descripteur LBP rotationnel pour un voisinage de 4, 8 et 16 sont effectués. Les résultats obtenus selon les deux métriques sont regroupés dans le tableau III.4. Les performances du descripteur en termes de robustesse par rapport aux attaques sont illustrées sur la figure III.13 pour une comparaison effectuée avec la distance MahCosine et la figure III.14 la distance χ^2 .

Résultats et Evaluation

Les métriques	MahCosine			χ^2		
Voisinage						
Attaques	P = 4	P = 8	P = 16	P = 4	P = 8	P = 16
Blur	66,67	70	76,67	60	66,67	80
Crop	73,33	80	86,67	86,67	90	90
Caption	83,33	80	83,33	76,67	73,33	90
Gamma	43,33	56,67	60	56,67	63,33	60
Noise	66,67	70	76,67	70	66,67	80
Compression	66,67	63,33	73,33	43,33	53,33	70
Hue	93,33	93,33	96,67	100	93,33	96,67
Flipping	96,67	93,33	93,33	3,33	86,67	93,33
Rotation	86,67	93,33	90	3,33	90	90

Tableau III. 4: Les résultats de LBP rotationnel pour un voisinage $P=4,8$ et 16.



Résultats et Evaluation

Figure III. 12: Les résultats de LBP rotationnel avec MahCosine pour un voisinage $P=4,8$ et 16.

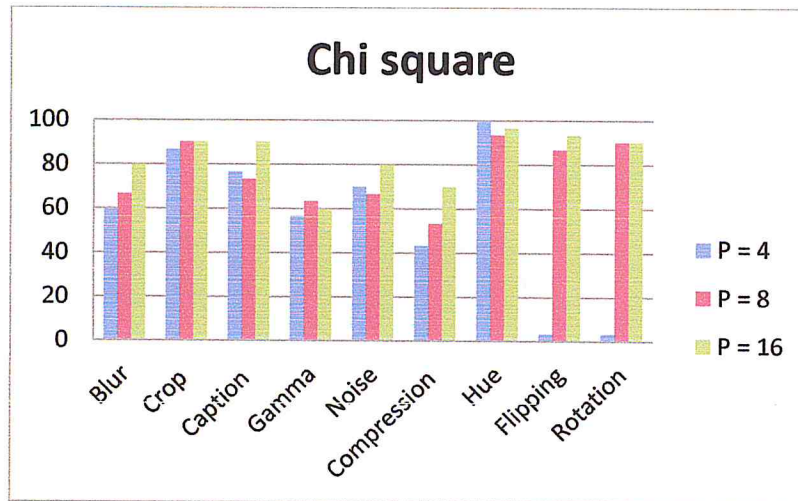


Figure III. 13: Les résultats de LBP rotationnel avec la métrique χ^2 pour un voisinage $P=4,8$ et 16.

D'après le tableau III.4 et les figures III.12 et III.13, nous nous déduisons que la métrique MahCosine avec un voisinage de $P=16$ donne les meilleures valeurs de taux de vérification dans la plupart des attaques « Blur », « Crop », « Caption », compression et « Noise ». Aussi, nous constatons que les taux de vérification des transformations rotation et « Flipping » sont augmentés par rapport aux résultats obtenus par les descripteurs LBP de base, LBP uniforme et LBP. D'où nous concluons que le descripteur LBP rotationnel pour un voisinage $P=16$ donne de meilleures performances avec MahCosine.

La figure III.14 illustre les courbes ROC des différentes attaques appliquées pour l'extraction des caractéristiques effectuée avec LBP rotationnel pour un voisinage $P=16$. la métrique utilisée est la distance MahCosine.

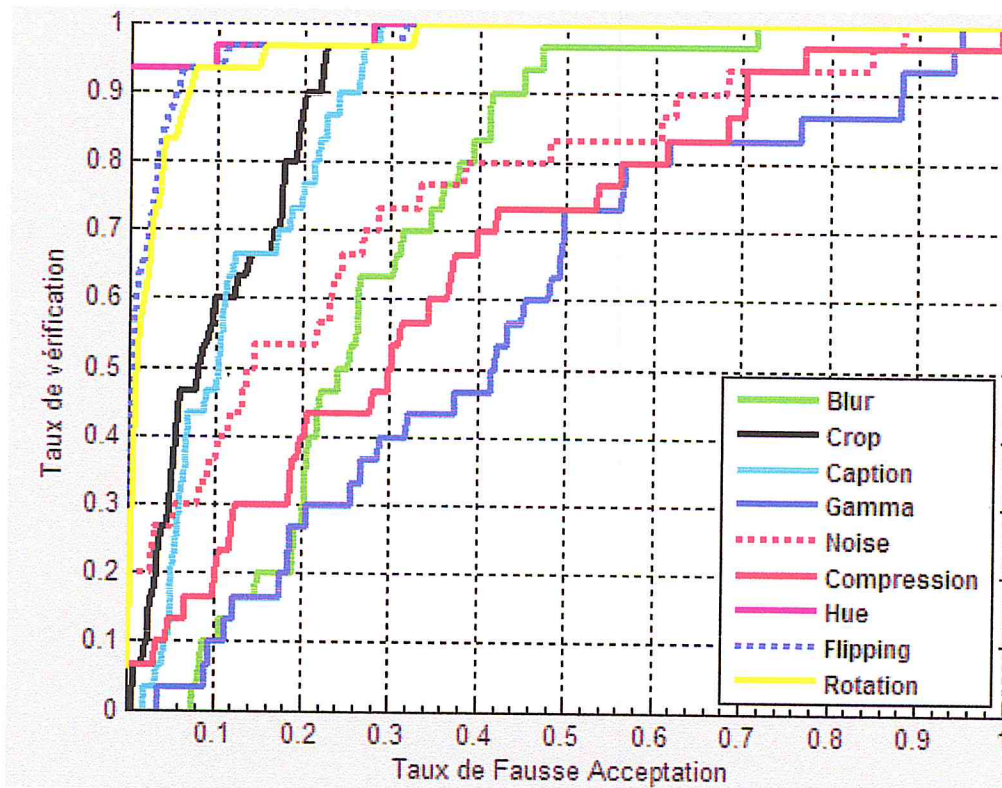


Figure III. 14: le ROC de différentes attaques avec LBP rotationnel, MahCosine et $P=16$.

Nous remarquons que LBP rotationnel avec un voisinage de $P=16$ est robuste à la modification de teinte (Hue) et aux transformations rotation et « Flipping ». En revanche, il montre une robustesse moindre par rapport aux transformations « Gamma » et compression.

II.1.4 Les résultats du descripteur LBP uniforme rotationnel

Les résultats obtenus dans le cadre du descripteur LBP uniforme rotationnel toujours en utilisant un voisinage de 4, 8 et 16 et les distances de comparaison sont regroupés dans le tableau III.5.

Résultats et Evaluation

Les métriques	MahCosine			Chi square χ^2		
	P = 4	P = 8	P = 16	P = 4	P = 8	P = 16
Voisinage						
Attaques						
Blur	66,67	60	70	53,33	40	56,67
Cropping	73,33	83,33	83,33	36,67	3,3333	83,33
Caption	83,33	76,67	70	36,67	46,67	43,33
Gamma	53,33	46,67	50	43,33	46,6667	50
Noise	66,67	66,67	66,67	30	86,6667	76,67
Compression	66,67	60	50	56,67	20	63,33
Hue	93,33	96,67	93,33	3,33	3,33	96,67
Flipping	90	80	86,67	30	83,33	36,67
Rotation	83,33	80	83,33	13,33	80	30

Tableau III. 5: Les résultats de LBP uniforme rotationnel.

Les performances du descripteur LBP uniforme rotationnel en termes de robustesse par rapport aux attaques sont illustrées sur la figure III.15 pour une comparaison effectuée avec la distance MahCosine et la figure III.16 la distance χ^2 .

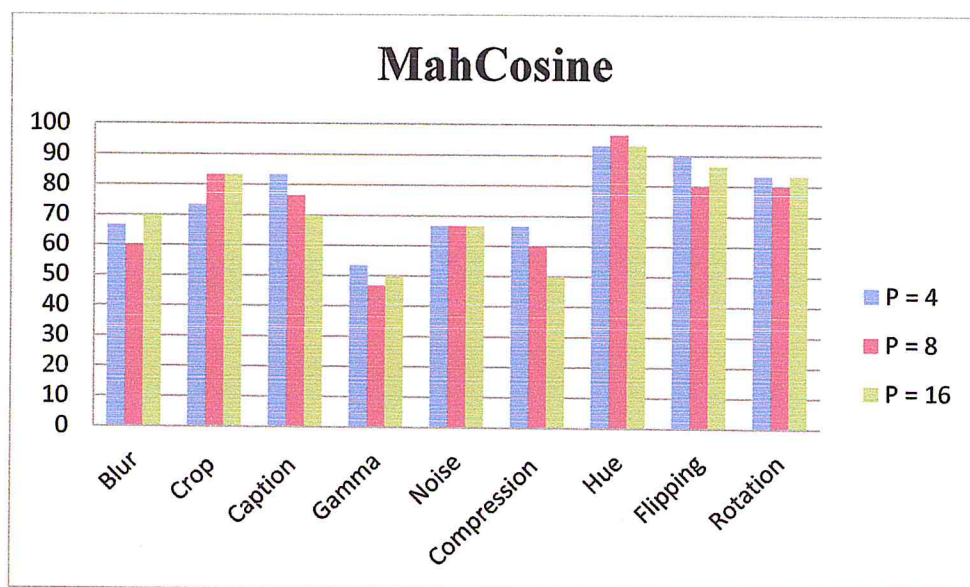


Figure III. 15: Les résultats de LBP uniforme rotationnel avec MahCosine.

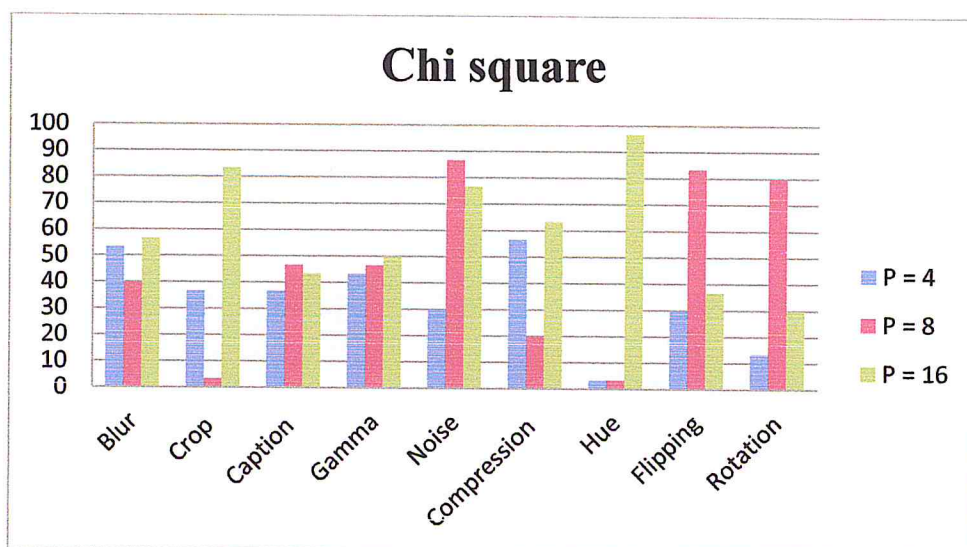


Figure III. 16: Les résultats de LBP uniforme rotationnel avec χ^2 .

En examinant les résultats regroupés sur le tableau III.5 et les ROC illustrés sur les figures III.15 et III.16, nous concluons que le descripteur LBP uniforme rotationnel pour un voisinage de $P=4$, donne un taux de vérification meilleur en utilisant la métrique MahCosine pour les attaques de « Caption », « Gamma », « Noise », « Compression » et « Rotation ». Par contre pour un voisinage égal à 8 et 16, le taux de vérification est élevé pour les attaques de « Blur », « Cropping » et « Hue ». D'où, nous concluons que LBP uniforme rotationnel donne de bonne performance en termes de robustesses pour un voisinage de 4 et la métrique MahCosine.

Résultats et Evaluation

De la même façon, que les descripteurs précédents, nous illustrons sur la figure III.17 les courbes les courbes ROC pour les différentes attaques appliquées pour l'extraction de caractéristiques par LBP uniforme rotationnel en utilisant MahCosine et un voisinage de $P=4$.

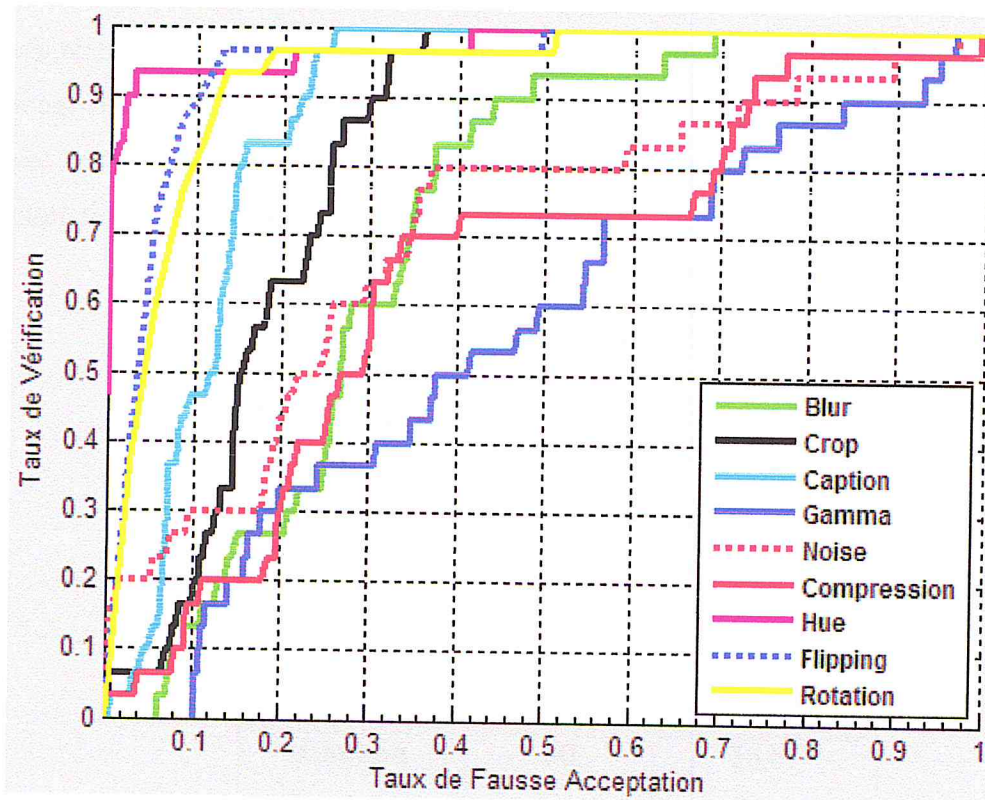


Figure III. 17: le ROC de différentes attaques avec LBP uniforme rotationnel, MahCosine et $P=4$.

D'après la figure III.17, nous constatons que LBP uniforme rotationnel avec un voisinage de $P=4$ est très robuste aux transformations de rotation et de Flipping et à la modification de teinte « Hue » et il est moins robuste à la transformation Gamma.

II.2 Performances du système dans le domaine spatio-temporel

Afin d'exploiter la composante temporelle de la vidéo, nous avons appliqué les descripteurs VLBP, VLBP rotationnel et LBP-TOP uniforme et non uniforme sur les deux types de vidéos à savoir les vidéos de références et les vidéos attaquées. Ces descripteurs sont appliqués pour les différents rayons et différents voisins.

Résultats et Evaluation

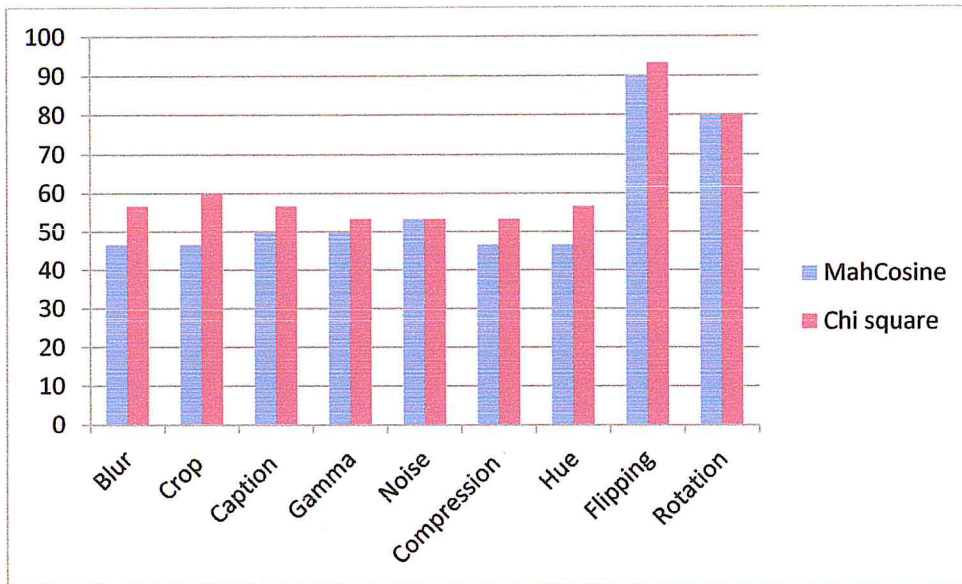


Figure III. 18: Les résultats de VLBP pour un voisinage $P=4$.

D'après le tableau III.6 et la figure III.18, nous déduisons que la métrique χ^2 donne des meilleurs valeurs de taux de vérification pour toutes les attaques. D'où le descripteur VLBP pour un nombre de voisins égale à 4 est robuste à toutes les attaques du moins avec un taux de vérification de 80% et 93,33% pour les attaques de rotation et de «Flipping» respectivement.

La figure III.19 illustre les courbes de taux de vérification pour les différentes attaques appliquées en employant le descripteur VLBP de base pour un nombre de voisins égale à 4 et la distance χ^2 .

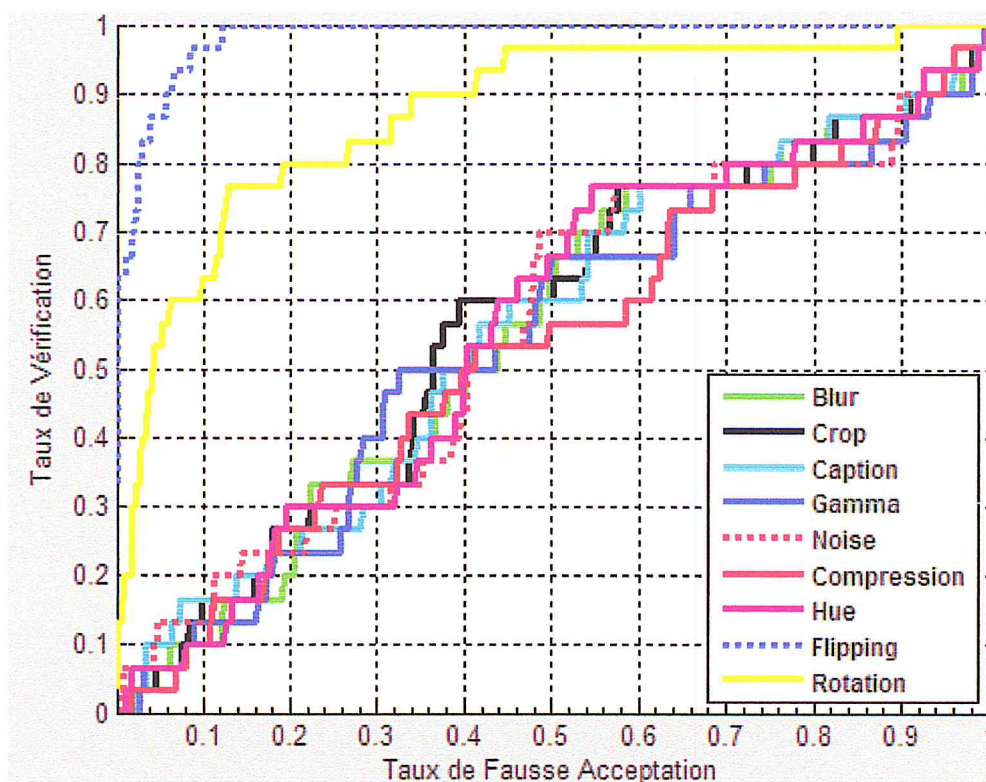


Figure III. 19: Les courbes ROC de différentes attaques avec VLBP pour un voisinage $P=4$.

D'après la figure III.19, nous constatons que VLBP avec un voisinage de $P=4$ est robuste aux attaques de rotation et « Flipping », par contre il est moins performant pour les autres attaques dont le taux de vérification est entre 53% et 60%.

II.2.2 Résultats obtenus par l'application du descripteur VLBP rotationnel

Les résultats obtenus dans le cadre de VLBP rotationnel pour un voisinage de 4 avec les deux métriques sont regroupés dans le tableau III.7, et illustrés sur la figure III. 20.

Résultats et Evaluation

Les métriques	MahCosine	Chi square χ^2
Attaques		
Blur	100	100
Cropping	96.67	100
Caption	96.67	100
Gamma	73.33	93.33
Noise	83.33	86.67
Compression	80	80
Hue	96.67	100
Flipping	93.33	96.67
Rotation	93.33	96.67

Tableau III. 7: Les résultats de VLBP rotationnel.

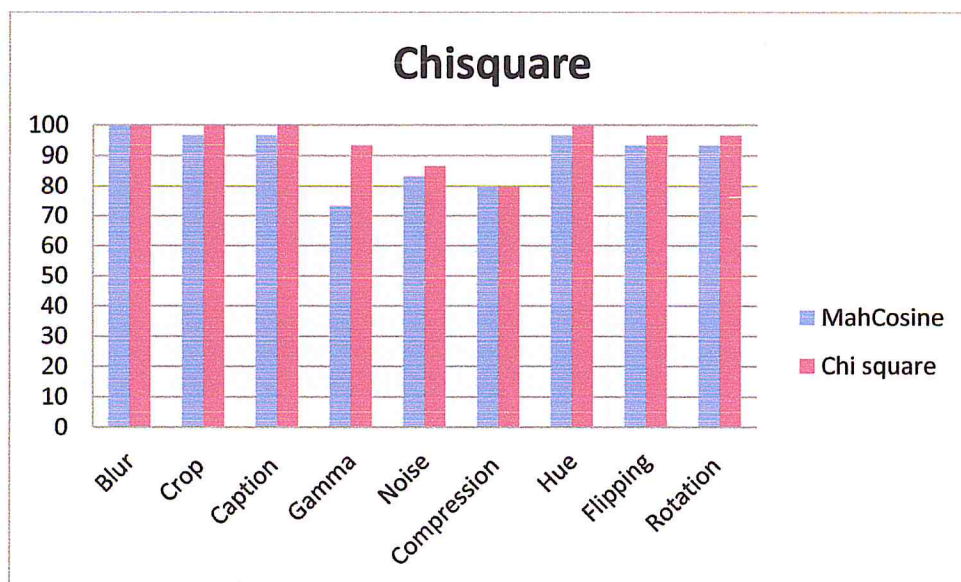


Figure III. 20: Les résultats de VLBP rotationnel avec les deux métrique.

Résultats et Evaluation

D'après le tableau III.7 et la figure III.20, nous constatons que le descripteur VLBP rotationnel pour un voisinage égale à 4, donne de bon taux de vérification pour la plupart.

La figure III.21 illustre les courbes de taux de vérification pour les différentes attaques appliquées en employant le descripteur VLBP rotationnel pour un nombre de voisins égale à 4 et la distance χ^2 .

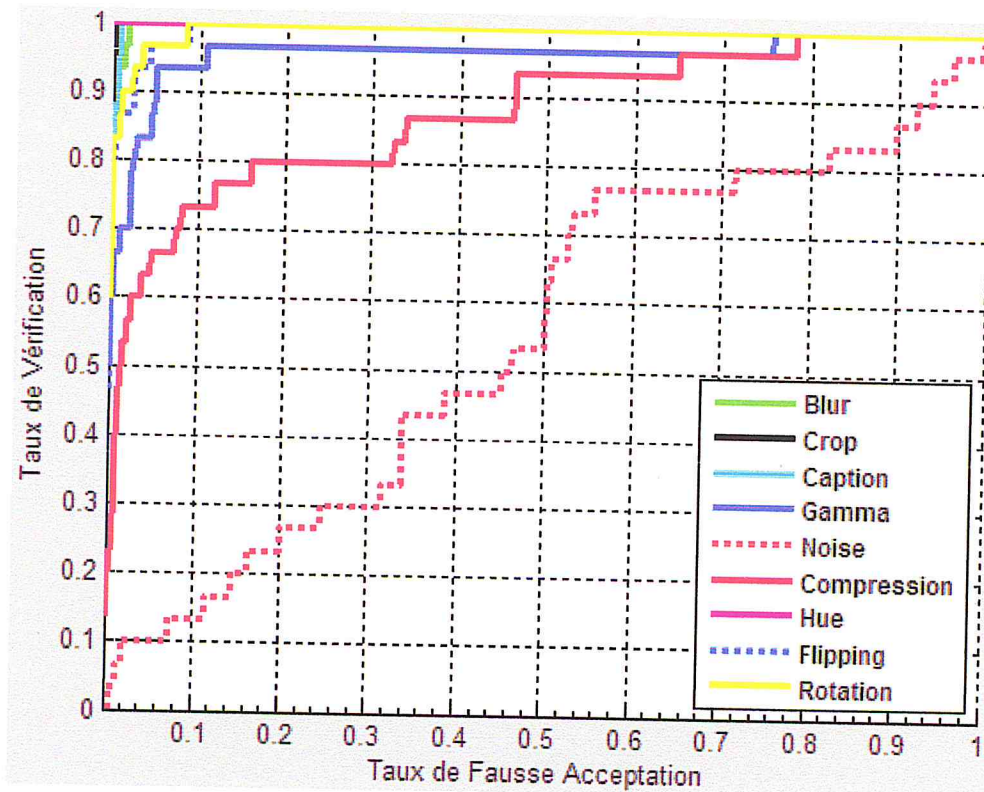


Figure III. 21: Les courbes ROC de différentes attaques avec VLBP rotationnel pour un voisinage $P=4$.

D'après la figure III.21, nous constatons que VLBP rotationnel avec un voisinage de 4, est robuste à la plupart des attaques, par contre il est moins robuste les attaques de bruit, « Noise », Gamma et de compression.

II.2.3 Résultats obtenus par l'application du descripteur LBP-TOP non uniforme.

Les résultats obtenus dans le cadre de LBP-TOP non uniforme pour un voisinage de 4, 8 et 16 avec les deux métriques sont regroupés dans le tableau III.8.

Les métriques Attaques	MahCosine			Chi square χ^2		
	P = 4	P = 8	P = 16	P = 4	P = 8	P = 16
Blur	60	60	60	93.33	96.67	93.33
Cropping	60	60	56.67	100	100	63.33
Caption	60	60	60	100	100	100
Gamma	60	60	56.67	70	70	63.33
Noise	60	60	60	73.33	76.67	83.33
Compression	56.67	56.67	60	86.67	86.67	83.33
Hue	60	60	60	96.67	96.67	96.67
Flipping	70	60	63.33	90	93.33	93.33
Rotation	66.67	66.67	56.67	80	76.67	76.67

Tableau III. 8: les résultats de LBP-TOP non uniforme.

Les performances du descripteur LBP-TOP non uniforme en termes de robustesse par rapport aux attaques sont illustrées sur la figure III.23 pour une comparaison effectuée avec la distance χ^2 et la figure III. 22 pour la distance MahCosine.

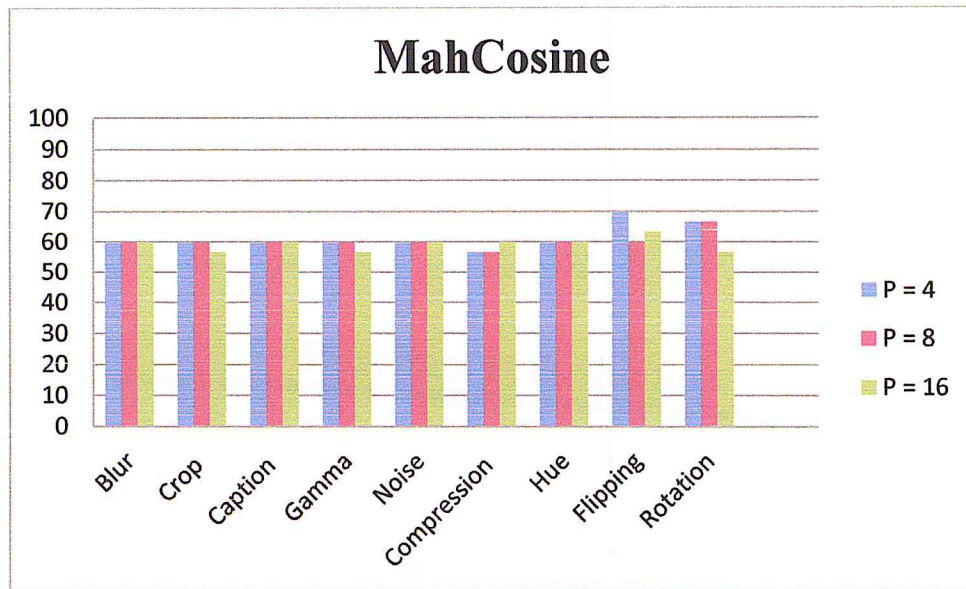


Figure III. 22: Les résultats de LBP-TOP non uniforme avec MahCosine.

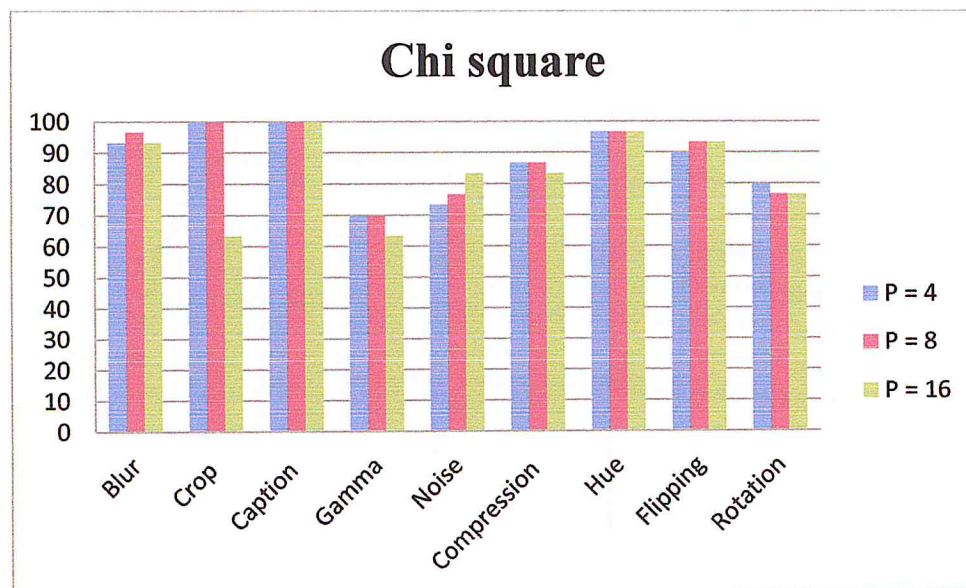


Figure III. 23: Les résultats de LBP-TOP non uniforme avec χ^2 .

D'après le tableau III.8 et les figures III.23 et III.22, nous retenons que le descripteur LBP-TOP non uniforme avec un voisinage de $P=8$ en utilisant Chi Square donne de meilleures performances en termes de taux de vérification dans la plus part des attaques. Ce qui conduit à dire que ce descripteur est robuste à toutes les attaques effectuées.

La figure III.24 illustre les courbes de taux de vérification pour les différentes attaques appliquées en employant le descripteur LBP-TOP non uniforme pour un nombre de voisins égale à 8 et la distance χ^2 .

Résultats et Evaluation

Les métriques	MahCosine			Chi square χ^2		
	TR	TR	TR	TR	TR	TR
Voisinage						
Attaque	P = 4	P = 8	P = 16	P = 4	P = 8	P = 16
Blur	93,33	93,33	86,67	96,67	100	100
Crop	99,33	100	93,33	99,85	100	100
Caption	100	100	96,67	100	100	100
Gamma	70	100	60	83,33	100	90
Noise	76,67	76,67	70	58,57	90	90
Compression	86,67	86,67	66,67	86,67	90	86,67
Hue	96,67	96,67	96,67	100	100	100
Flipping	90	93,33	93,33	90	96,67	96,67
Rotation	80	76,67	76,67	80	76,67	73,33

Tableau III. 9: Les résultats de LBP-TOP uniforme.

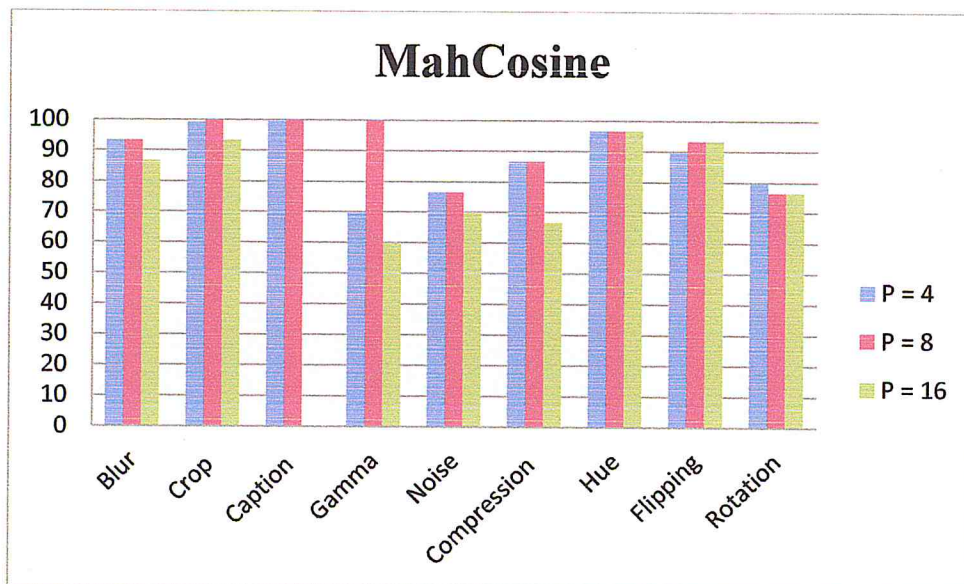


Figure III. 25: Les résultats de LBP-TOP uniforme avec MahCosine.

Résultats et Evaluation

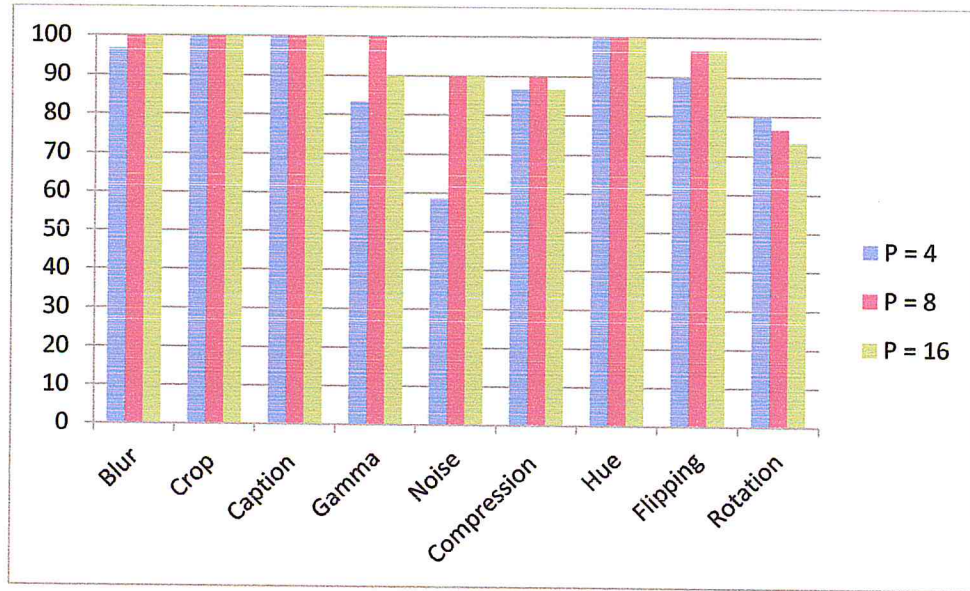


Figure III. 26: Les résultats de LBP-TOP uniforme avec χ^2 .

D'après le tableau III.9 et les deux figures III.25 et III.26, nous déduisons que LBP-TOP uniforme avec un voisinage $P=8$ et la comparaison effectuée avec la distance Chi Square donne des meilleurs taux de vérification dans la plus part des attaques sauf pour les attaques de rotations dont la valeur est 76,67%.

La figure III.27 illustre les courbes de taux de vérification pour les différentes attaques appliquées en employant le descripteur LBP-TOP uniforme pour un nombre de voisins égale à 8 et la distance χ^2 .

Résultats et Evaluation

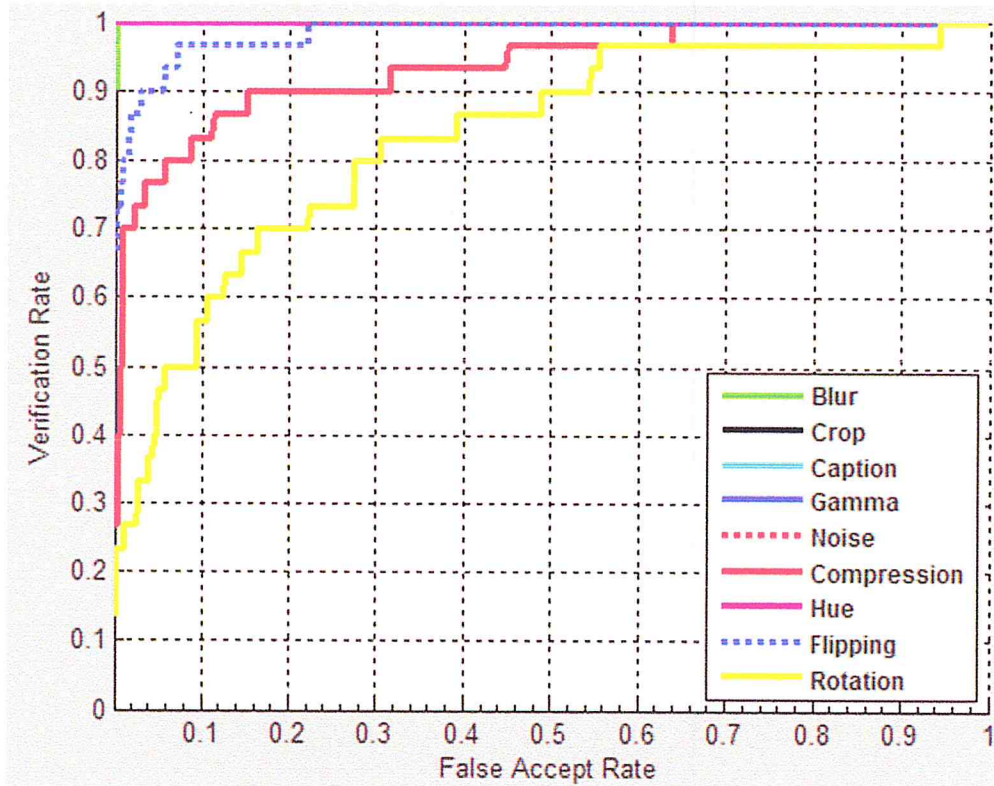


Figure III. 27: Les courbes ROC de différentes attaques avec LBP-TOP uniforme, χ^2 .

D'après la figure III.27, nous constatons que LBP-TOP uniforme avec un voisinage de $P=8$ est très robuste à la majorité des transformations sauf l'attaque de rotation qui moindre.

Après avoir appliqué le descripteur LBP et ses variantes dans le domaine spatial, nous pouvons confirmer que dans un système de détection de copie vidéo, les descripteurs LBP spatiaux avec la métrique MahCosine donne de bonne performances. Le problème qui surgit est que la caractéristique temporelle n'est pas prise en considération ce qui provoque une défaillance dans les systèmes de détection de copie vidéo pour les vidéo comprenant beaucoup de mouvement. Pour les extensions spatio-temporelles du descripteur LBP, nous remarquons que le descripteur LBP_TOP uniforme est robuste à la majorité des attaques sauf aux manipulations de rotation et de Flipping. La figure III.27 illustre les taux obtenus avec le meilleur descripteur, le meilleur voisinage en prenant en compte le contenu spatial et spatio-temporel.

Résultats et Evaluation

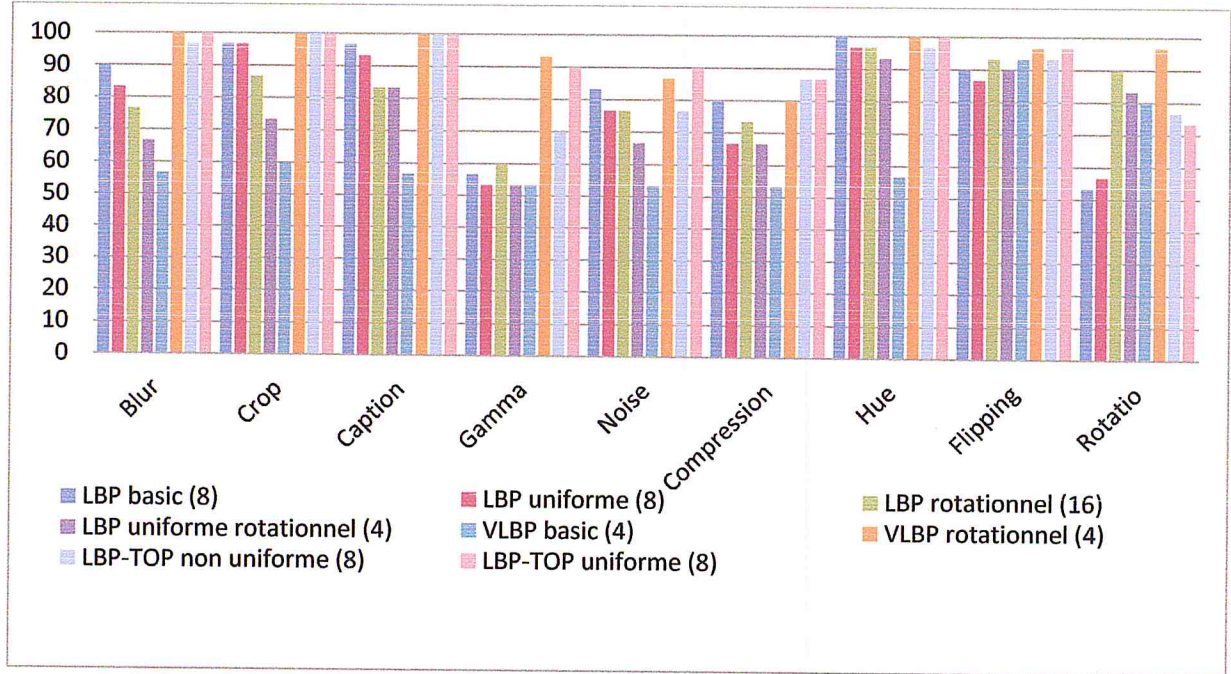


Figure III. 28: Comparaison des performances des différents descripteurs LBP et ses variantes.

D'après la figure III.28 et les tests effectués, nous déduisons que le descripteur LB-TOP uniforme est robuste à toutes les transformations sauf à la Rotation. Pour résoudre ce problème, et afin d'améliorer la robustesse du système à l'attaque de rotation, nous avons introduit dans notre système de détection de copie vidéo un prétraitement qui consiste à appliquer l'algorithme log polaire [66] sur les images de la séquence vidéo en entrée. Son principe consiste à transformer les coordonnées des pixels de l'image du domaine c

cartésienne (x,y) au domaine log-polaire. Les coordonnées polaires (ρ, θ) correspondent respectivement à la distance radiale du centre et de l'angle sont définies comme suit.

$$(\rho, \theta) = (\log(\rho), \theta) \quad (3.2)$$

$$\rho = \sqrt{(X - X_c)^2 + (y - y_c)^2} \quad \text{et} \quad \theta = \tan^{-1}\left(\frac{y-y_c}{x-x_c}\right) \quad (3.3)$$

Où X_c et y_c sont les coordonnées du centre de l'image log-polaire.

Après l'application de l'algorithme log polaire, le taux de vérification de LBP-TOP par rapport à la transformation de rotation est passé de 76,67% à 90%.

II.2.5 Comparaison des résultats

Afin d'évaluer notre système de détection de copie vidéo basé sur le contenu, nous avons comparé la méthode développée à celle proposée dans la littérature [67] et celles développées au CDTA et parues dans [40] [41]. Le tableau III.10 illustre les résultats des taux de vérification obtenus pour les attaques en commun.

Attaques	Ren et al. [67]	Block-BSIF [41]	ring-BSIF [41]	Color-BSIF-ring [68]	LBP-TOP uniforme
Caption	100	100	99	98.11	100
Gamma	100	100	99	98.15	100
Blur	87	93	93	95.05	100
Crop	100	100	98	98.65	100
Rotation	82	77	93	95.88	90
Flipping	85	80	97	98.73	96.67

Tableau III. 10: comparaison des résultats de base de données TRECVID 2009.

Nous remarquons que les résultats obtenus par l'application du descripteur LBP-TOP uniforme sont meilleurs pour les transformations « Caption », Gamma, « Blur » et « Cropping ». Il reste à améliorer les performances du système par rapports aux transformations rotation et « Flipping ». Ceci en introduisant la caractéristique couleur.

III. La décision

Le but de cette partie est de détecter le type d'attaque de la vidéo requête. Pour cela les vidéos attaquées sont classées deux groupes copie et non copie. Cette classification est basée sur la mesure de similarité χ^2 . La vidéo appartient à la classe « copie » s'il existe son origine dans la base de test sinon elle est non copie.

Résultats et Evaluation

Pour classifier les vidéos attaquées, nous avons proposé dans un premier temps l'utilisation de l'apprentissage automatique (Machine learning). Pour cela, nous avons appliqué la méthode des arbres de décision [69] [70] pour un apprentissage supervisé. Les arbres de décision sont des outils d'aide à la décision représentant des situations plus ou moins complexes à lesquelles on doit faire face sous la forme graphique d'un arbre. L'extrémité de chaque branche est une feuille présentant le résultat obtenu en fonction des décisions prises à partir de la racine de l'arbre.

III.1 Application des arbres de décision dans le processus de détection

L'utilisation des arbres de décision dans la détection consiste à tester une multitude d'algorithmes d'arbre pour évaluer le résultat de modélisation de la décision « copie » ou « non copie ». Pour effectuer ce traitement, nous avons utilisé le logiciel WEKA [71] développé pour les plateformes d'apprentissage artificiel. Notre choix pour le logiciel WEKA s'appuie sur :

- Sa simplicité : utilisation de l'interface graphique
- Sa capacité d'analyser des résultats avec l'évaluation de performances.
- Les nombreux algorithmes d'arbre de classification qu'il contient.

Avec le logiciel WEKA, nous avons abouti à des valeurs inférieures à 45% en termes de précision. L'échantillon de test est la diagonale de la matrice de comparaison de chaque attaque. La raison pour laquelle, nous avons proposé une autre méthode basée sur le seuil de détection [72].

III.2 Application Le seuil de détection

Le principe de cette méthode est comme suit [72]: si la distance de la vidéo attaquée V1 est égal ou inférieur au seuil de détection, nous prenons comme « copie » la vidéo V1, sinon la vidéo V1 est considérée « non copie ». Le seuil est calculé de la façon suivante :

$$seuil = (Max_s + Min_s)/2 \quad (3.5)$$

Où

$$Max_s = Max (max (TS), min(FS)) \quad (3.6)$$

$$Min_s = Min (max (TS), min(FS))$$

Conclusion

Dans ce chapitre, Nous avons présenté les différents résultats obtenus par l'application des descripteurs locaux LBP et ses variantes selon plusieurs voisinages. Après une comparaison des performances en termes de robustesse par rapport aux attaques utilisées, nous avons adopté le descripteur spatio-temporel LBP-TOP uniforme avec un voisinage égal à 8 dans le système (CBCD) en raison de sa robustesse à la majorité des attaques. Une amélioration de la robustesse de ce descripteur par rapport à la rotation est effectuée. Celle-ci est réalisée par l'introduction de l'algorithme log polaire a été effectuée pour rendre le descripteur spatio-temporel plus robuste à l'attaque de rotation. La prise de décision sur la vidéo si elle est copie ou non est basé sur le seuil de détection.



Chapitre IV

Implémentation et Capteurs d'écran

Introduction

Une fois que nous avons comparé l'intégralité des descripteurs LBP et ses variantes. Nous avons opté pour un système de détection de copie vidéo basé sur les les caractéristiques spatio-temporelle.

Dans cette partie, nous allons concevoir une interface graphique à notre application pour permettre à un n'importe quel utilisateur d'utiliser notre application, D'où L'objectif de cette étape est de décrire de façon détaillée et précise le système de détection, ainsi que la présentation des interfaces graphiques sous la forme d'une capture d'écran.

I. Les Langages et les Outils Utilisée

L'implémentation des différentes procédures s'est faire sous environnement Windows en utilisant la connexion entre la programmation orienté objet basée sur java et le langage de calcul Matlab.



I.1 le langage Matlab

MATLAB [73] est un langage de manipulation numérique de matrices, il est une boîte à outils d'analyse numérique, avec un environnement agréable pour faire du calcul numérique avec sa programmation matricielle. Il dispose des méthodes usuelles de l'analyse numérique:

1. résolution de système linéaire
2. détermination des valeurs propres
3. résolution de problème non-linéaire
4. résolution d'équations différentielles (raides)
5. intégration numérique
6. optimisation et contrôle
7. graphique 2D et 3D



I.2 Le langage Java

Développé par Sun Microsystems depuis la fin des années 1980, Java est un langage de programmation à usage général, évolué et orienté objet dont la syntaxe est proche du C. Il existe deux types de programmes en Java : les applets et les applications. Une application autonome (stand alone program) est une application qui s'exécute sous le contrôle direct du système d'exploitation. Une applet est une application qui est chargée par un navigateur et qui est exécutée sous le contrôle d'un plugin de ce dernier.



I.3 Eclipse

Appelé aussi Eclipse Project ou encore Eclipse Platform. La caractéristique essentielle d'Eclipse est l'extensibilité de l'environnement. Plus que de se focaliser sur un environnement de développement Java, les concepteurs d'Eclipse se sont efforcés avant tout de créer un socle applicatif sur lequel viennent se greffer des modules et/ou plugins.

Eclipse est un environnement de développement intégré (EDI), placé en open source par Sun. En plus de Java, Eclipse permet également de supporter différents autres langages, comme Python, C, C++, JavaScript, XML, Ruby, PHP et HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web).

Eclipse constitue par ailleurs une plateforme qui permet le développement d'applications spécifiques (bibliothèque Swing (Java)). L'IDE Eclipse s'appuie sur cette plateforme.

I.4 API (Application programming interface) utilisées

Les API utilisé pour l'implémentation de notre application présentes les caractéristiques suivantes.



I.4.1 Matlabcontrol

Est une API de JMI (Java-to-Matlab Interface), qui permet d'appeler Matlab à partir de Java. Elle prend en charge l'appel Matlab de deux façons différentes: le contrôle local où le code Java est lancé à partir de Matlab, et le contrôle à distance où le code Java lance Matlab.



I.4.2 Java Media Framework (JMF)

Est une API unifiée pour la synchronisation, le traitement et l'affichage de données temporelles comme les données audio et de vidéo en java, développé par Sun, Silicon Graphics et Intel. Java Media Player est la première étape et comporte la synchronisation, l'exécution, la présentation et le stockage de données temporelles compressées. Elle support plusieurs formats comprennent l'UA, AVI, MIDI, MPEG, QuickTime, et WAV avec plusieurs utilisations, telle que: la transmission de l'audio et de vidéo en temps réel sur Internet. Diffusion de programmes de radio ou de télévision en direct lecture de différents fichiers multimédias dans une application Java lecture de streaming multimédia sur Internet.

III. La simulation et les captures d'écran

Après lancement de l'interface sous l'éditeur Eclipse nous constatons la fenêtre principale qui regroupe plusieurs modules chacun devant répondre à une opération précise,

La figure IV.1 représente la fenêtre principale de notre application.

Implémentation et Capteurs d'écran

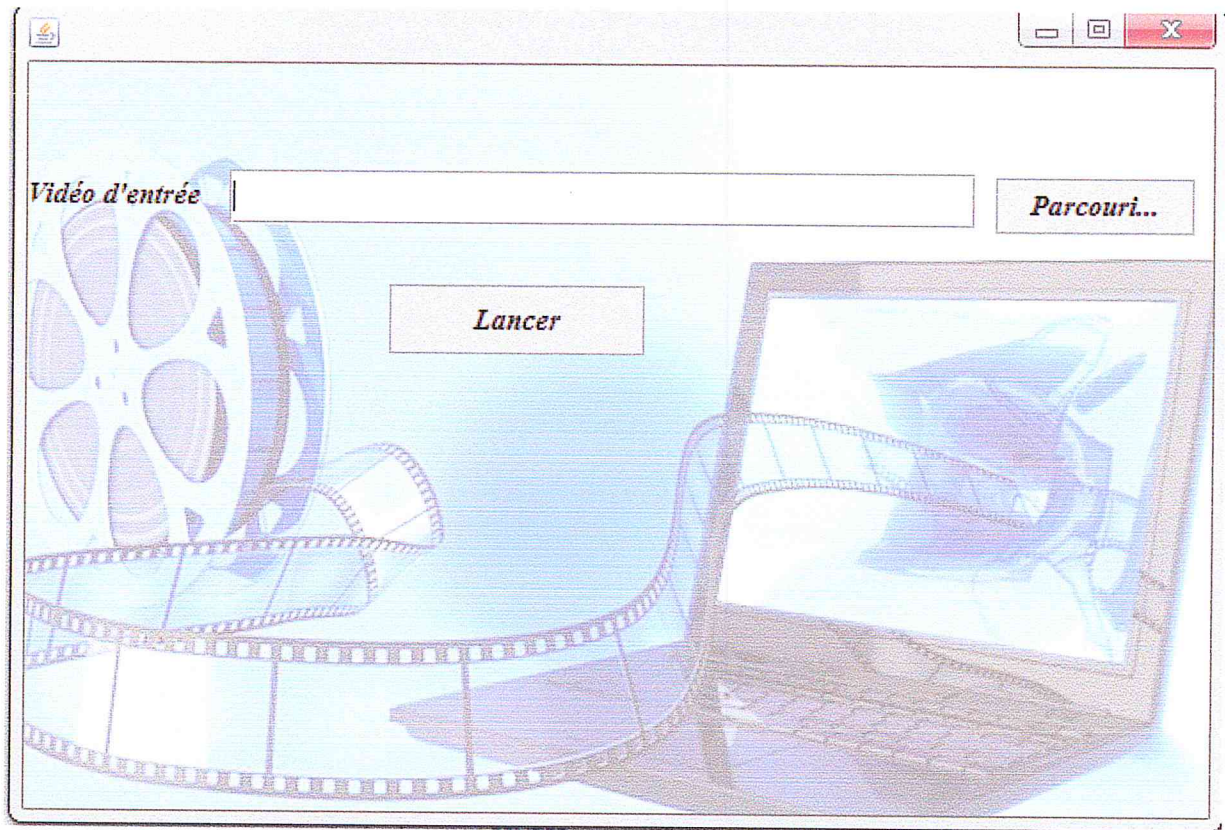


Figure IV. 1: la fenêtre principale de notre application.

Le bouton « parcourir » permet à l'utilisateur de rechercher l'emplacement de la vidéo d'entrée.

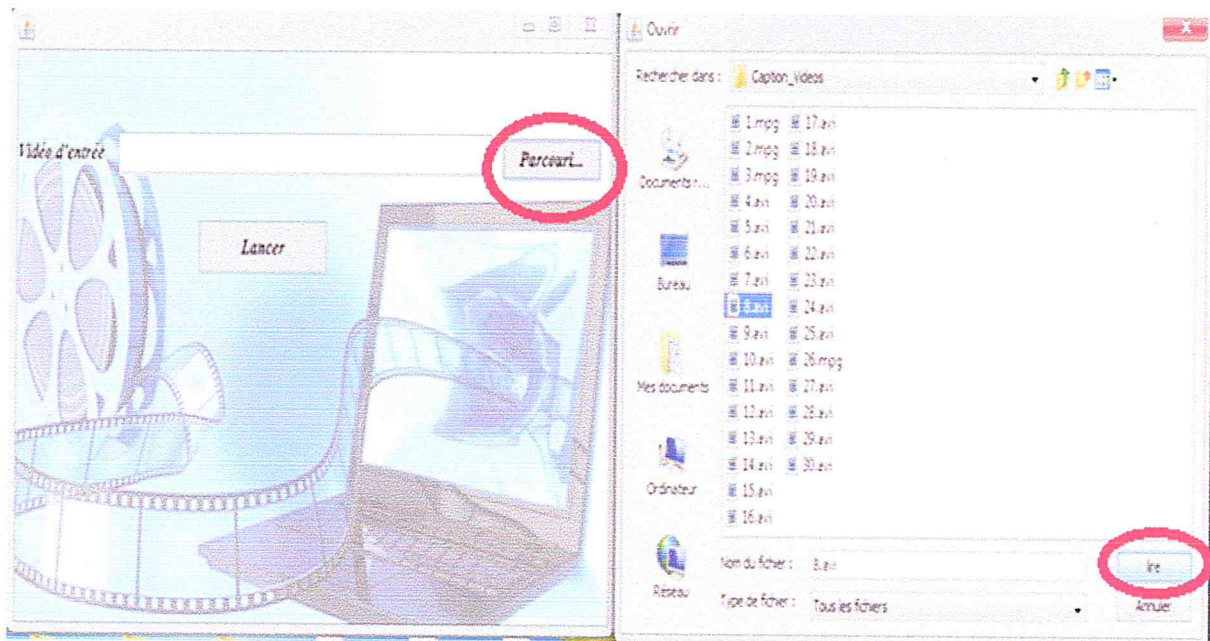


Figure IV. 2: Recherche d'une vidéo d'entrée.

Implémentation et Capteurs d'écran

Le bouton « lire » permet de lire la vidéo d'entrée sélectionnée, comme elle est représentée dans la figure IV.2.

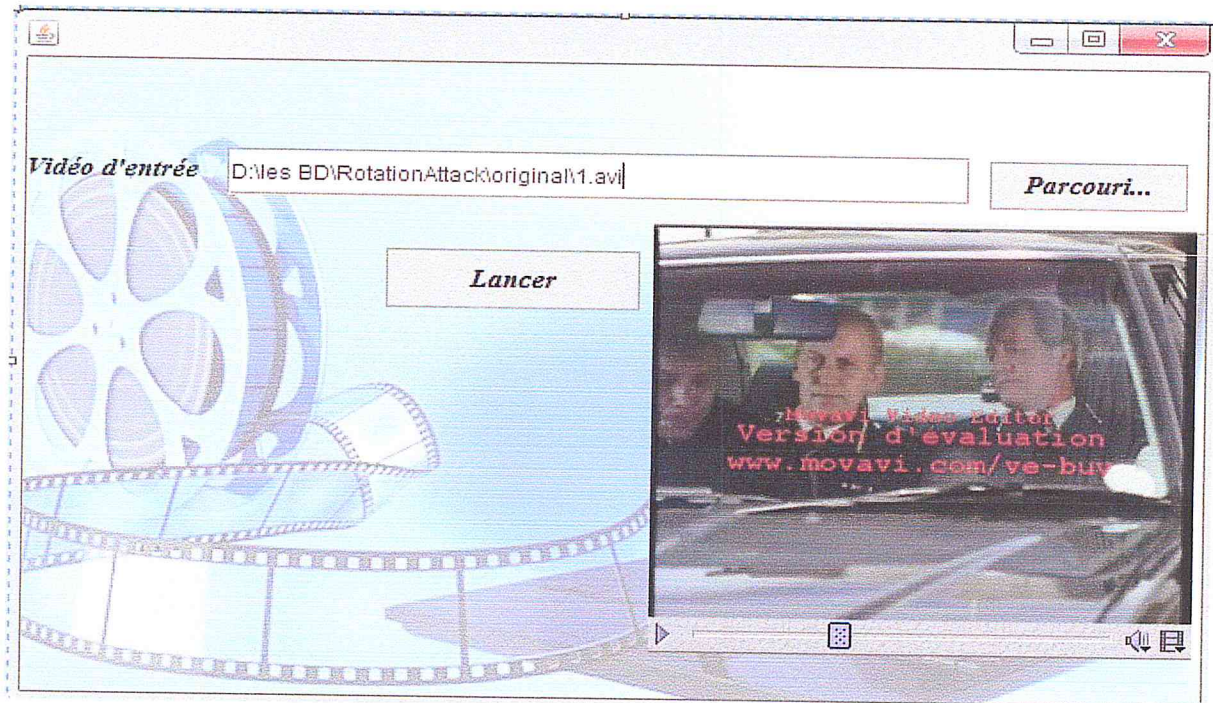


Figure IV. 3: lecture de la vidéo d'entrée.

Le bouton « lancer » permet de connecter Java avec Matlab afin de déclencher le processus de détection de copie vidéo sous Matlab.

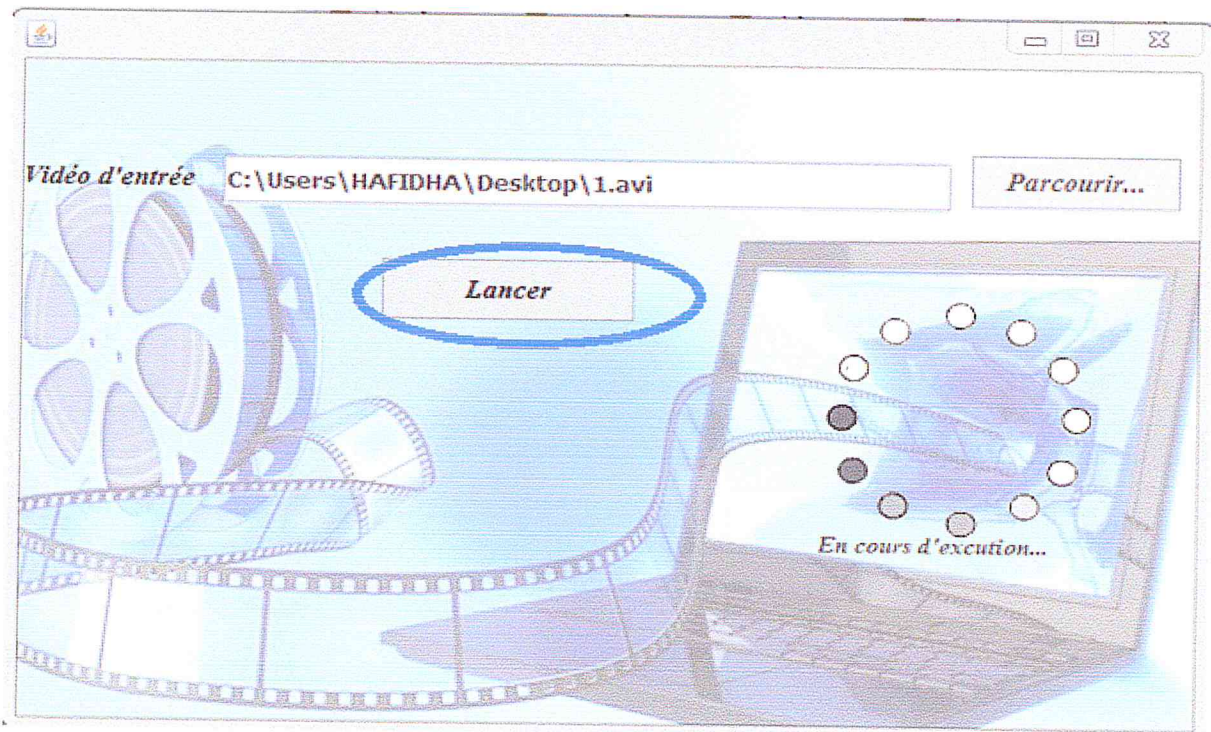


Figure IV. 4: Lancement de processus de détection.

Implémentation et Capteurs d'écran

La fenêtre suivante IV.5 illustre les résultats de processus de détection de copie vidéo à savoir le vecteur caractéristiques de la vidéo d'entrée, le vecteur de comparaison et la décision.

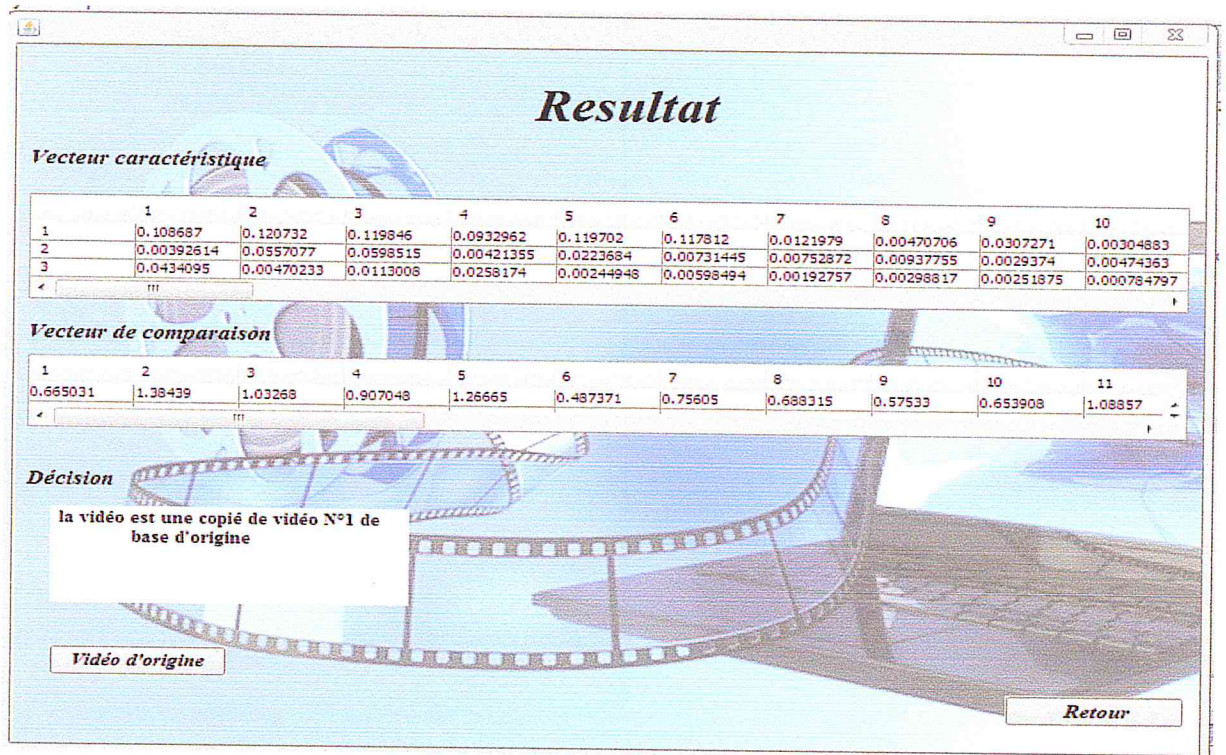


Figure IV. 5: La fenêtre de résultats.

Le bouton « vidéo d'origine » permet de visualiser la vidéo d'origine de la vidéo d'entrée.

Le bouton « Retour » passe à l'interface principale.

Implémentation et Capteurs d'écran

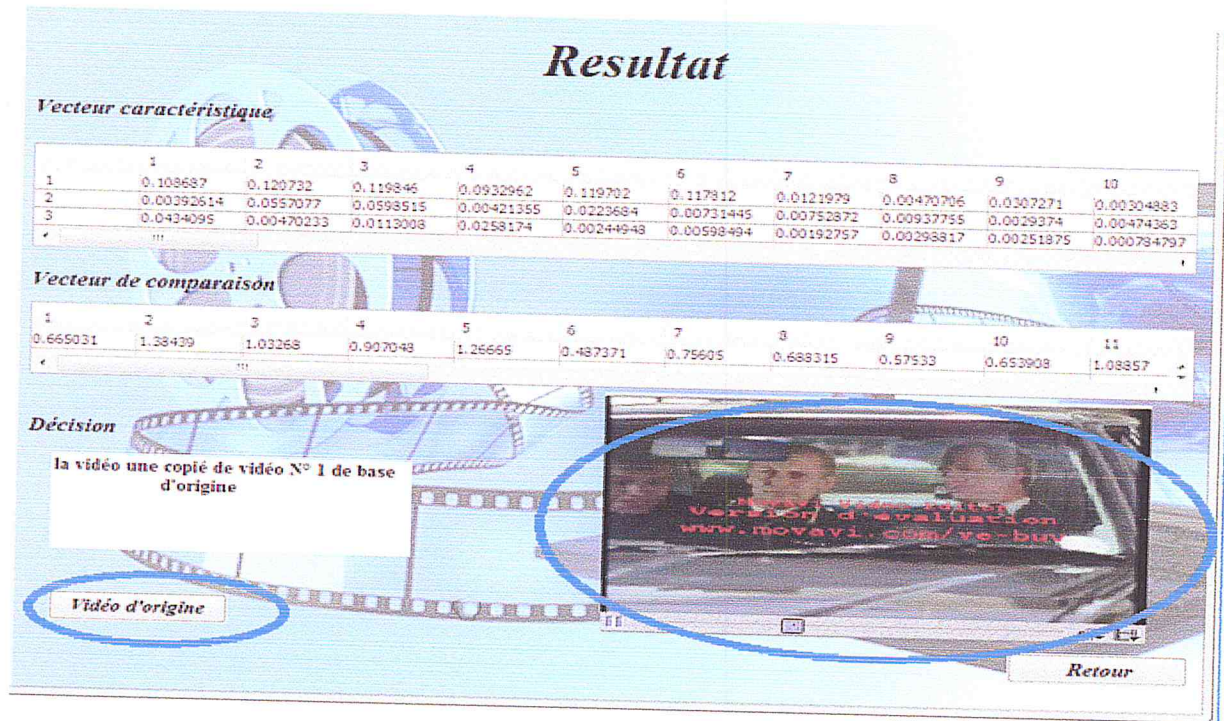


Figure IV. 6: visualisation de la vidéo d'origine dans l'interface.

Conclusion

Nous avons présenté dans ce chapitre l'implémentation de notre système en décrivant les outils et langages utilisés pour la mise en œuvre des différents composants du système. Nous avons utilisé principalement une connexion entre les langages Java et Matlab et l'environnement de développement Eclipse pour présenter les entrées et les résultats de notre système tandis que les traitements sont faits sous Matlab. Enfin, nous avons présenté quelques-unes des fenêtres principales de notre application.

Conclusion général

Notre but a été de concevoir un système de détection de copie vidéo basé sur le contenu spatio-temporel qui soit facile et robuste au maximum de manipulations de traitement d'images telles que la rotation, le flipping, l'introduction du bruit, le flou et etc. Les descripteurs locaux semblent les outils les plus performants qui représentent le contenu vidéo sous format de vecteurs caractéristiques discriminants.

Nous avons présenté dans ce mémoire les résultats de l'application du descripteur local LBP et ses variantes pour la détection de copie vidéo. Le descripteur LBP-TOP uniforme permet de mieux représenter le contenu vidéo. Les résultats obtenus ont montré son efficacité en termes de robustesses aux attaques ciblées. Bien que ce descripteur a marqué une faiblesse dans la robustesse par rapport à la rotation des images. Afin d'améliorer cette dernière, nous avons introduit un prétraitement sur images de la séquence vidéo. Il consiste à appliquer le log polaire sur toutes les images de la séquence avant de procéder à l'extraction des caractéristiques. Ce prétraitement a amélioré les performances du système réalisé. Néanmoins il reste des perfectionnements à faire qui feront l'objet des travaux de recherche futurs. Parmi celles-ci, nous pouvons citer :

- Prendre en compte d'autres caractéristiques de la vidéo telle que la composante couleur.
- Améliorer la robustesse de la rotation, en proposant d'autres algorithmes.
- Tester la plate forme réaliser avec un voisinage plus grand $P=24$ qui nécessite plus d'espace mémoire.
- Améliorer le module de décision afin de détecter la nature de l'attaque de la vidéo copie.

- [83] M.Pietikäinen, A.Hadid, G.Zhao and T.Ahonen, Computer Vision Using Local Binary Patterns.
- [84] T.Ojala, T.Pietikäinen and M.Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* 24(7), pp 971–987, 2002.
- [85] R.Chetverikov and D. Péteri, A brief survey of dynamic texture description and recognition. In: *Proc. International Conference on Computer Recognition Systems*, pp 17–26 2005.
- [86] Y.Zhao, G.Huang, X.Gizatdinova and M.Pietikäinen, Combining dynamic texture and structural features for speaker identification. In: *Proc. ACM Multimedia Workshop on Multimedia in Forensics, Security and Intelligence*, pp. 93–98 (2010).
- [87] M.Zhao and G.Pietikäinen, Dynamic texture recognition using volume local binary patterns. In: *Dynamical Vision. Lecture Notes in Computer Science*, vol. 4358, pp. 165–177. Springer, Berlin (2007).
- [88] T.Ojala, T.Pietikäinen and M.Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* 24(7), pp 971–987 2002.
- [89] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikainen, X. Chen, W. Gao, WLD: A Robust Local Image Descriptor, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32 , no.9, 2010.
- [90] K. W. Bowyer, K. Chang and P. Flynn. A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition. *Computer Vision and Image Understanding*, vol. 101, pp.1-15, 2006.
- [91] T. Ojala, M. Pietikäinen, and D. Harwood, A comparative study of texture measures with classification based on featured featured distributions. *Pattern Recognition*, 29(1), pp 51–59, 1996.

Avancées.Alger, Algérie,2015.

[69] BREIMAN L , FRIEDMAN J , OLSHEN R , STONE C., Classification and Regression Trees, California: Wadsworth Internationa, 1984.

[70] Quinlan , J. Ross, Induction of decision trees. Machine Learning, 1986..

[71] <http://www.cs.waikato.ac.nz/ml/weka/>.

[72] Y.Himeur, Karima Ait Sadi, Joint Color and Texture Descriptor Using Ring Decomposition for Robust Video Copy Detection in Large Databases, Centre de d'veloppement des Technologies Avancées.Algérie, 2015.

[73] « <http://www.mathworks.com>,» [En ligne].

[74] Allinson., J. Li and, comprehensive review of current local features for computer A comprehensive review of current local features for computer, June 2008.

[75] A. Hadid, M. Pietikäinen, and T. Ahonen, A discriminative feature space for detecting and recognizing faces, In CVPR (2), pages 797–804, 2004.

[76] Schmid, K. Mikolajczyk and C, A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(10), pp 1615 –1630, October 2005.

[77] Bremond, P. Bilinski and F, Evaluation of local descriptors for action recognition in videos. In Proceedings of the 8th international conference on Computer vision systems ICVS'11, pp 61–70, Berlin, Heidelberg, 2011.

[78] S. Gauglitz, T. Höllerer, and M. Turk, Evaluation of interest point detectors and feature descriptors for visual tracking. Int. J. Comput. Vision, 94(3) PP 335–360, September 2011.

[79] J. Kannala and E. Rahtu, BSIF: Binarized Statistical Image Features. 21st International Conference on Pattern Recognition (ICPR), pp.1363-1366, 2012.

[80] A. Hyvärinen et al. Natural Image Statistics. Springer, 2009.

[81] E. Tola, V. Lepetit, and P. Fua, A fast local descriptor for dense matching. IEEEConference on Computer Vision and Pattern Recognition, pp 1- 8, 2008.

[82] T. Ahonen, A. Hadid, and M. Pietikainen, Face description with local binary patterns: Application to face recognition.IEEE Trans. Pattern Anal. Mach. Intell., 28(12), pp 2037–2041, 2006.

- [55] 5. DAISY refE. Tola, V. Lepetit, and P. Fua. A fast local descriptor for dense matching. IEEEConference on Computer Vision and Pattern Recognition, pages 1-8, 2008..
- [56] M. P. a. D. H. T. Ojala, A comparative study of texture measures with classification based on featured featured distributions. Pattern Recognition, 29(1):51–59, 1996..
- [57] A. H. a. M. P. T. Ahonen, Face description with local binary patterns: Application to face recognition.IEEE Trans. Pattern Anal. Mach. Intell., 28(12):2037–2041,, 2006..
- [58] A. H. G. Z. T. A. Matti Pietikäinen, Computer Vision Using Local Binary Patterns.
- [59] Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. 24(7),971–987 (2002).
- [60] Chetverikov, D., Péteri, R.: A brief survey of dynamic texture description and recognition. In: Proc. International Conference on Computer Recognition Systems, pp. 17–26 (2005).
- [61] Zhao, G., Huang, X., Gizatdinova, Y., Pietikäinen, M.: Combining dynamic texture and structural features for speaker identification. In: Proc. ACM Multimedia Workshop on Multimedia in Forensics, Security and Intelligence, pp. 93–98 (2010).
- [62] Zhao, G., Pietikäinen, M.: Dynamic texture recognition using volume local binary patterns. In: Dynamical Vision. Lecture Notes in Computer Science, vol. 4358, pp. 165–177. Springer, Berlin (2007).
- [63] Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. 24(7), 971–987 (2002).
- [64] Trecvid conference website. <http://trecvid.nist.gov/>..
- [65] A.OUAMANE, Reconnaissance Biométrique par Fusion Multimodaledu Visage 2D et 3D, Doctorat en sciences en Electronique, Université Mohamed Khider – Biskra, 11 Juin 2015.
- [66] S.Saikiran Thunuguntla, B.Tech et G.Rajam, OBJECT TRACKING USING LOG-POLAR TRANSFORMATION, India, August, 2005.
- [67] J. Ren, F. Chang, T. Wood. Efficient Detection via Aligning Video Signature Time Series. In Proceedings of the 2nd ACM Int. Conf. on Multimedia Retrieval, no. 14. 2012.
- [68] Y.Himeur Joint Color and Texture Descriptor Using Ring Decomposition for Robust Video Copy Detection in Large Databases, Centre de d'veloppement des Technologies

- [41] Geert Willems, Tinne Tuytelaars, Luc Van Gool, Spatio-Temporal Features for Robust Content-Based Video Copy Detection, MIR'08, Vancouver, Canada, 2008.
- [42] Y. Himeur, K. Ait-Sadi, A rotation/flipping invariant BSIF descriptor for video copy detection using a ring decomposition. Submitted to 7th Int. Conf. Soft Computing Pattern Recognition (SoCPaR), Fukuoko, Japan, 2015.
- [43] J. L. a. N. M. Allinson., A comprehensive review of current local features for computer A comprehensive review of current local features for computer, June 2008..
- [44] M. P. a. T. A. A. Hadid, A discriminative feature space for detecting and recognizing faces., In CVPR (2), pages 797–804, 2004. .
- [45] K. M. a. C. Schmid, A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(10):1615 –1630, October 2005.
- [46] P. B. a. F. Bremond, Evaluation of local descriptors for action recognition in videos. In Proceedings of the 8th international conference on Computer vision systems ICVS'11, pages 61–70, Berlin, Heidelberg, 2011. Springer-Verlag.
- [47] T. H. o. a. M. T. S. Gauglitz, Evaluation of interest point detectors and feature descriptors for visual tracking. Int. J. Comput. Vision, 94(3):335–360,, September 2011.
- [48] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikainen, X. Chen, W. Gao, WLD: A Robust Local Image Descriptor, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32 , no.9, 2010..
- [49] J. Kannala and E. Rahtu. BSIF: Binarized Statistical Image Features. 21st International Conference on Pattern Recognition (ICPR), pp.1363-1366, 2012..
- [50] K. W. Bowyer, K. Chang and P. Flynn. A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition. Computer Vision and Image Understanding, vol. 101, pp.1-15, 2006..
- [51] A. Hyvärinen et al. Natural Image Statistics. Springer, 2009..
- [52] Shervin Rahimzadeh Arashloo and Josef Kittler, MemberClass-Specific Kernel Fusion of Multiple Descriptors for Face Verification Using Multiscale Binarised Statistical Image Features IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL.9 DECEMBER 2014.
- [53] G. David Lowe, Distinctive Image Features from Scale-Invariant Keypoints, University of British Columbia, Canada, January 5, 2004.
- [54] J. Koenderink, The structure of images. Biological Cybernetics, pages 363-396. 1984.

- [28] C.Herley, *Extracting Repeats from Media Streams, Acoustics, Speech, and Signal Processing*, 2004.
- [29] S.Outara, *stratégie de segmentation d'image multi-composantes par analyse d'histogrammes multidimensionnels*, These de doctorat, école doctorale d'angers STIM, France. 2009.
- [30] M.Lezoray Olivier, *segmentation d'image couleur par morphologie mathématique et classification de données par réseaux de neurones:application à la classification de cellules en cytologie des séreuses*,thèse de dovtorat,2000, université de CAEN/BASSE-NORMANDIE,france.
- [31] N. Morizet, «N.Morizet, *Reconnaissance Biométrique par Fusion Multimodale du Visage et de l'Iris*,domain other. Télécom ParisTech, 2009».
- [32] O. Küçüktunç, U. Güdükbay, O. Ulusoy, *Fuzzy color histogram-based video segmentation*, *Computer Vision and Image Understanding* 114 (1), pp 125–134, 2010.
- [33] S. Gauglitz, T. H ollerer, and M. Turk, *Evaluation of interest point detectors and feature descriptors for visual tracking*. *Int. J. Comput. Vision*, 94(3), pp 335–360, September 2011.
- [34] N. Ejaza, T.Bin Tariq, W.Sung, *Adaptive key frame extraction for video summarization using an aggregation mechanism*, 2012.
- [35] A.Bhatt, S. Lee, C.D. Yoo, *Robust video fingerprinting for content-based video identification*, *IEEE Trans. Circ. Syst. Video Technol.* p 983–988,18 (7) 2008.
- [36]] M. R. Naphade, M. M. Yeung, and B. Yeo. *Novel scheme for fast and efficient video sequence matching using compact signatures*. 3972(1) pp 564–572, 1999.
- [37] Indyk, P., Iyengar, G. and Shivakumar, N., *Finding pirated video sequences on the Internet*, Tech. Rep., *Stanford InfoLab*, Stanford University, Feb. 1999.
- [38] Shivakumar, N., *Detecting digital copyright violations on the Internet*, Ph.D. Dissertation, Stanford University, Aug. 1999.
- [39] M. Malekesmaeili, M. Fatourech, and R. K. Ward, *Video copy detection using temporally informative representative images*, in *Proc. Int Conf.Machine Learning and Applications* , pp. 69-74, Dec. 2009.
- [40] I. Simand, D. Pellerin, S. Bres and J. M. Jolion, *SPATIO-TEMPORAL SIGNATURES FOR VIDEO COPY DETECTION'* *Advanced Concepts for Intelligent Vision systems*, Bruxelles. pp. 421-427. 2004.

- [14] S. Cheung and A. Zakhor, «Estimation of Web Video Multiplicity, In Proceedings of the SPIE - Internet Imaging, vol. 3964, pp. 34-46. San Jose, California January 22-28, 2000.
- [15] H. Chang, S. Sull, and S. Lee, indexing scheme for content-based retrieval, IEEE Trans. Circuits Syst. Video Technol Efficient video.
- [16] Y.Rui,A.she and T.Huang, Modified fourier descriptors for shape representation-a practical approach In proc.offnt work shop on image database and multimedia search.
- [17] Y.Zhuang and y.pan, Content video similaritymodel.In proc of ACM int conf.on multimedia ,page 465-467, 2000.
- [18] S.CHEROUATI, Développement d'outils numérique pour l'amélioration et l'analyse d'image, Mémoire de DUEA en Informatique, USDB,2009.
- [19] <http://reussirsinformationvideo.com/les-categories-de-format-video/>.
- [20] R.Mohan, Video sequence matching, in: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP98),vol. 6, pp. 3697–3700, 1998.
- [21] C. Cheung and A. Zakhor, Efficient video similarity measurement with video signature, Proceedings of the 9th IEEE International Conference on Image Processing, vol 1, pp. 621-624, Sept. 2002.
- [22] C.Hong Hoi, W.Wang, R.Michael, L.yu, A Novel Scheme for Video Similarity Detection. CIVR 2003.
- [23] G.Molina, N. Shivakumar, Finding nearreplicas of documents on the web, in World Wide Web and Databases. International Workshop WebDB'98, Valencia, Spain, pp. 204-12, Mar 1998.
- [24] A.Border, S.Glassman, M.Manasse, and G.Zweig, Syntactic clustering of the web, in Proc. 6th Int. World Wide Web Syntactic clustering of the web, in Proc 6th Int. World web, Conf., vol. 29, no. 8–13, Computer Networks and ISDN Systems, pp. 1157–1166 Sept. 1997.
- [25] A. Hampapur and R. M. Bolle, Comparison of distance measures for video copy detection, In Proc. of Int. Conf. on Multimedia and Expo, Aug.2001.
- [26] S. Cheung and A. Zakhor, Estimation of Web Video Multiplicity, In Proceedings of the SPIE - Internet Imaging, vol. 3964, pp. 34-46. San Jose, California January 22-28, 2000, ..
- [27] H. Chang, S. Sull, and S. Lee, indexing scheme for content-based retrieval, IEEE Trans. Circuits Syst. Video Technol Efficient video., vol. 9.

Bibliographie

- [1] S. Poullot , M. Crucianu _2 and O. Buisson Passage à l'échelle de la fouille vidéo basée sur la détection de copies Institut National de l'Audiovisuel, 94366 Bry-sur-Marne, France.
- [2] S.Satoh, News video analysis based on identical shot detection, in Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'02), pp. 69–72, 2002.
- [3] F.Medjahed, Détection et Suivi d'Objets en Mouvement, Mémoire de Magister Département d'Electronique, Université des Sciences et de la Technologie d'Oran U. S. T. O. 2012.
- [4] S.Lahlou, L'activité du point de vue de l'acteur et la question de l'intersubjectivité [huit années d'expériences avec des caméras miniaturisées fixé au front des acteurs entre subcams. Communications, 80(1), 2006.
- [5] « <http://www.tutorials-computer-software.com/2012/12/definition-plan-cinema-audiovisuel.html>,» [En ligne].
- [6] A.Saoudi, Empreinte numérique et indexation de la vidéo, 2007.
- [7] M.A.Bourenane, un outil pour l'indexation des vidéos personnelle par le contenu université de québec à 03 rivières, Thèse de doctorat 2009.
- [8] S.Lefevre, Détection d'événements dans une séquence vidéo. Université François Rabelais-Tours, 2002.
- [9] O.Adjemout, Reconnaissance automatique de formes à partir des paramètres morphologiques, de couleur et de texture : application au tri des grains de semences, thèse de magister, UNMTO, 2005.
- [10] S.Bedouhene, recherche d'image par contenu, mémoire de magister, Université Mouloud Mammeri, Tizi-ouzo.
- [11] Y.W, Y.Zhuang and y.pan, Content video similarity model. In proc of ACM int conf.on multimedia, page 465-467, 2000.
- [12] N.Rabia, revue de méthodes de segmentation d'images texturées cas des images couleur, 2011.
- [13] A. Hampapur and R. M. Bolle, Comparison of distance measures for video copy detection, In Proc. of Int. Conf. on Multimedia and Expo, 2001.