MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
UNIVERSITE ABDELHAMID IBN BADIS - MOSTAGANEM

Faculté des Sciences Exactes et de l'Informatique
Département de Mathématiques et d'Informatique
Filière : Informatique

MEMOIRE DE FIN D'ETUDES
Pour l'Obtention du Diplôme de Master en Informatique
Option : **Ingénierie des Systèmes d'Information**

THEME :

Détection d'un cut dans une vidéo surveillance en utilisant les techniques du data mining.

Etudiant : « BENTATA Kada Omar »

Encadrant : « Henni Karim Abdelkader »

Année Universitaire 2016/2017

Abstract

In today's digital era, there are large volumes of long-duration videos resulting from movies, documentaries, sports and surveillance cameras floating over internet and video databases (YouTube). Since manual processing of these videos are difficult, time-consuming and expensive, an automatic technique of abstracting these long-duration videos are very much desirable.

In this paper, we will see how to realize a cut detection in a long video for its indexing using the visual characteristic extraction tool SIFT

Keywords : Image indexing, Video indexing, SIFT, CBIR, CBVR, Segmentation.

Résumé

Dans l'ère numérique d'aujourd'hui, il existe de volume énorme de vidéos de longue durée qui sont le résultat de films, de documentaires, de sport et de caméras de surveillance flottant sur Internet et les bases de données vidéo (ex. YouTube). Comme le traitement manuel de ces vidéos est difficile et coûteux, une technique automatique d'indexation de ces vidéos de longue durée est très souhaitable.

Dans ce travail, Nous allons voir comment on peut faire la détection des coupures dans une vidéo longue pour l'indexer en utilisant le moyen d'extraction des caractéristiques visuelles SIFT.

Mot clés : Indexation des images, Indexation des vidéos, SIFT, CBIR, CBVR, échantillonnage, Segmentation.

Remerciement

Tout d’abord, louange à « Allah » qui m’a guidé sur le droit chemin tout au long du travail et m’a inspiré les bons pas et les justes reflexes. Sans sa miséricorde, ce travail n’aura pas abouti.

Ce projet de fin d'étude s'est déroulé au sein de l'université AbdelHamid Ibn Badis à Mostaganem.

En préambule à ce mémoire, je souhaite adresser mes remerciements les plus sincères aux personnes qui ont apporté leur aide et ont contribué à l'élaboration de ce mémoire ainsi qu'à la réussite de cette formidable année universitaire.

Je tiens à remercier sincèrement Monsieur « Henni Karim Abdelkader», qui, en tant qu'encadreur, s'est toujours montré à l'écoute et très disponible tout au long de la réalisation de ce mémoire, ainsi pour l'inspiration, l'aide et le temps qu'il a bien voulu me consacrer et sans qui ce mémoire n'aurait jamais vu le jour. « Merci pour tous ses conseils, ses persévérances et ses bienveillances »

Je saisie l’occasion pour remercier vivement tout le corps professoral et administratif de la Faculté des Sciences Exactes & Informatique.

Mes remerciements aussi aux membres de jury pour l'honneur qu'ils ont fait de bien vouloir analyser ce travail et apporter leurs suggestions.

Il m’est particulièrement agréable d’exprimer ma reconnaissance et mess vifs remerciements à mes parents, qui ont toujours su me faire confiance et me soutenir sans compter dans mes études

Et enfin, j’adresse une pensé toute particulière à tout ceux qui ont, un jour ou l’autre, croisé mon chemin, que ce soit l’université ou n’importe où dans ce petit monde.

Dédicaces

Je dédie ce modeste travail

A mes très chers parents, pour leur soutien ; mon père qui s'est sacrifie afin que rien n'entrave le déroulement de mes études, ma mère qui n'a pas cessé de prier pour moi et de m'encourager dans les moments difficiles

- *A mon cher frère*
- *A mes chères sœurs*
- *A toute la famille*
- *A tous mes amis et touts mes collègues de la promotion d'Informatique (2016 /2017).*

Merci à tous.

LISTE DES MATIERES

LISTE DES FIGURES

Introduction générale

La vidéosurveillance est un système de caméras et de transmission d'images, disposé dans un espace public ou privé pour le surveiller à distance ; il s'agit donc d'un type de télésurveillance. Les images obtenues avec ce système, peuvent être traitées automatiquement et/ou visionnées puis archivées ou détruites. La surveillance a pour but de contrôler les conditions de respect de la sécurité, de la sûreté ou de l'exécution d'une procédure particulière.

L'installation de systèmes de vidéosurveillance sont multiples, toutefois la sécurité publique ainsi que la protection des biens mobiliers ou immobiliers font office d'éléments phares dans la justification de la vidéosurveillance.

Les vidéosurveillances sont de plus en plus communes, générant des milliers d'heures de vidéos archivées tous les jours. Ces données sont rarement traitées en temps réel et principalement utilisées pour les enquêtes sur les scènes après les événements.

Il est extrêmement difficile pour l'humain de parcourir tout le contenu d'une vidéo archivée a fin de trouver ce qu'il veut, il risque de perdre énormément de temps précieux sans résultat, c'est pour cela il faut une technique automatique en vue de faciliter la tâche et traiter les vidéos archivées ou bien directement les flux vidéos.

Une solution à ce problème consiste à marquer les moments quintessentiels d'une manière périodique ou bien en temps réel et les enregistrer sous données facilement consultables. Par des techniques d'extractions de données d'un flux vidéo.

Cette solution est basée sur comment détecter les coupures de la vidéo qui donnent des segments généralement peu différents entre eux.

Le but de notre travail est de chercher les points de coupures dans une vidéosurveillance, ces dernières constitueront les limites des segments quand il y a un changement majeur dans la cène.

Afin de décrire le travail effectué, ce mémoire comporte les quatre principaux chapitres organisés comme suit :

Le premier chapitre portera des définitions sur les concepts des systèmes de l'indexation et la recherche et des notions générale sur les vidéos, cela nous aide en suite à mieux comprendre le point de coupure entre deux plan.

Dans le deuxième chapitre nous expliquerons le rôle et les types des outils utilisés pour l'extraction de l'information du contenu d'une vidéo, et nous détaillerons quelques types qui ont une réputation dans ces domaines.

Le troisième chapitre est une idée générale sur les différents aspects de la conception de notre projet et les méthodes choisies et utilisées dans notre approche, il détaille en plus les techniques employées théoriquement.

Le quatrième chapitre est consacré à la présentation de la mise en œuvre, l'implémentation et la performance notre application.

Chapitre I : La vidéo et son indexation par le contenu

I.1. Introduction

Nous sommes confrontés à une explosion de l'information numérique selon des entreprises spécialisées dans les études de marché sur les technologies de l'information qui annoncent que la quantité d'informations numériques créées, capturées et reproduites en 2006 était de 161 milliards de giga-octets, donc environ de trois millions de fois l'information dans tous les livres écrits auparavant. [19]

Dans ce chapitre on va s'intéresser sur la nature de la donnée vidéo et comment il est possible de l'indexer en basant sur son contenu.

I.2. La donnée vidéo

Les vidéos sont composées de deux parties une partie sonore et une partie vidéo (visuelle) c'est deux parties sont synchronisés.

On va s'intéresser à la partie vidéo car généralement la vidéosurveillance ne prend pas en considération la partie sonore. La partie vidéo c'est une séquence d'images appelées frames ou trames.

Les frames sont affichés dans une cadence qui dépasse la capacité de l'œil humain à observer qu'il y a une sorte de retardement.

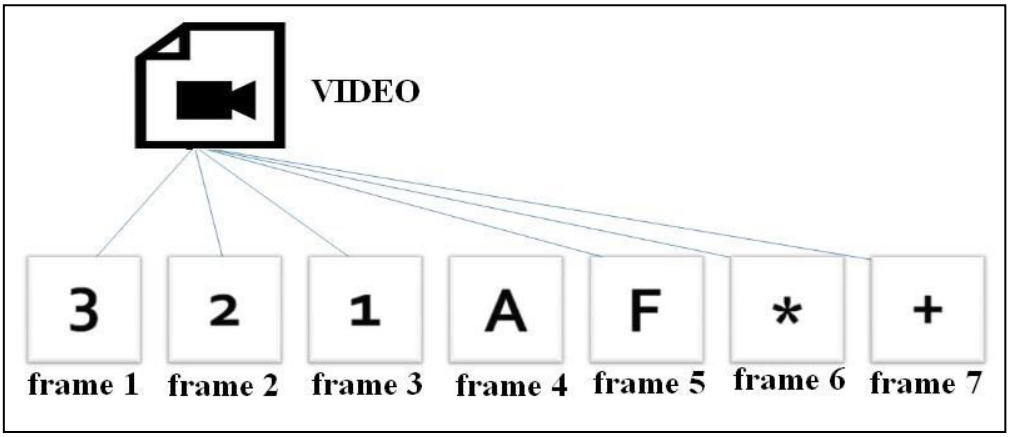


Fig.1. la donnée vidéo et les frames

I.2.1. Le frame

Le frame est une image numérique simple. L'affichage de ces images avec fluidité donne l'impression de regarder une vidéo.[12]

Cette fluidité est définie par le nombre de frames affichés par l'unité de temps **fps** (*frame per seconde).* Généralement 30 fps est la norme dans nos jours, même que récemment la fluidité peut dépasser 120 fps.

I.3. L'indexation et la recherche

I.3.1. L'indexation

L'indexation consiste à extraire, représenter et organiser efficacement le contenu des documents d'une base de données. [7]

le Dictionnaire encyclopédique de l'information et de la documentation édité par Calcaly (2001) , définit l'indexation par sa finalité ne notant que :

> « L'indexation a pour but de faciliter l'accès au contenu d'un document ou d'un ensemble de documents à partir d'un sujet ou d'une combinaison de sujets (ou de tout autre type d'entrée utile à la recherche). Cela s'applique aussi bien a l'élaboration des index situés généralement en fin d'ouvrage qu'à l'usage des langages documentaires pour analyser le contenu d'une collection de documents et permettre par la suite, grâce aux fichiers ou à la banque de données ainsi alimentée, la recherche des informations répondant à une préoccupation particulière ».

I.3.2. L'indexation par le contenu

Maintenant nous savons bien que la vidéo n'est qu'un ensemble d'images (*frames*), donc l'indexation et la recherche serait faite sur des images numérique, voila pourquoi nous devons aborder la façon de manipuler, traiter et indexer des images par leurs contenu.

Le fait d'indexer d'images par leur contenu est de les représenter par des signatures numériques (*descripteurs*) comme l'illustre la figure .2. (la partie en bas). Cette opération est réalisée en deux étapes :

- *l'analyse et l'extraction*
  Il s'agit d'analyser les images pour en extraire l'essentiel, comme de capturer les couleurs ou les textures caractéristiques, l'existence ou l'inexistence d'un objet ou bien l'identification des visages. [7]

Dans la dernière partie, les descripteurs doivent être stockés d'une manière organisée afin d'optimiser la recherche.

I.3.3. La recherche par mots clés

Les premiers systèmes de recherche d'images utilisaient des mots-clés associés aux images pour les caractériser. Grâce à cette association de mots-clés, il suffit d'utiliser les méthodes basées sur le texte pour retrouver les images contenant les mots-clés. Plusieurs moteurs de recherche proposent ces recherches d'images basées sur le texte. Ils s'appuient sur le principe simple que dans une page web, il y a une forte corrélation entre le texte et les images présentes.

Le principal problème de ces recherches par mots-clés est que le résultat peut être complètement hors sujet.

L'association de textes à l'image est une démarche réaliste pour de petites bases de données (taille inférieure à 10 000 images), mais il est complètement impensable pour de grandes bases de données (nombre d'images supérieur à 10 000). En effet, le temps passé à l'association de mots-clés et la pertinence des mots-clés restent très subjectifs et très dépendants des personnes qui effectuent l'association.

I.3.4. La recherche par le contenu

La recherche par le contenu consiste à rechercher les images en n'utilisant que l'image elle même sans aucune autre information.

Nous avons dit qu'en recherche par mot clé l'utilisateur doit construire une requête textuelle, mais la procédure est bien difficile pour les images et encor pour les vidéos.

Dans la recherche d'image par son contenu, la requête construite par l'utilisateur est une image qui sera transformée à une signature similairement a l'étape de l'indexation. cette dernière sera comparée aux signatures déjà existante dans la base de donnée (figure 2).

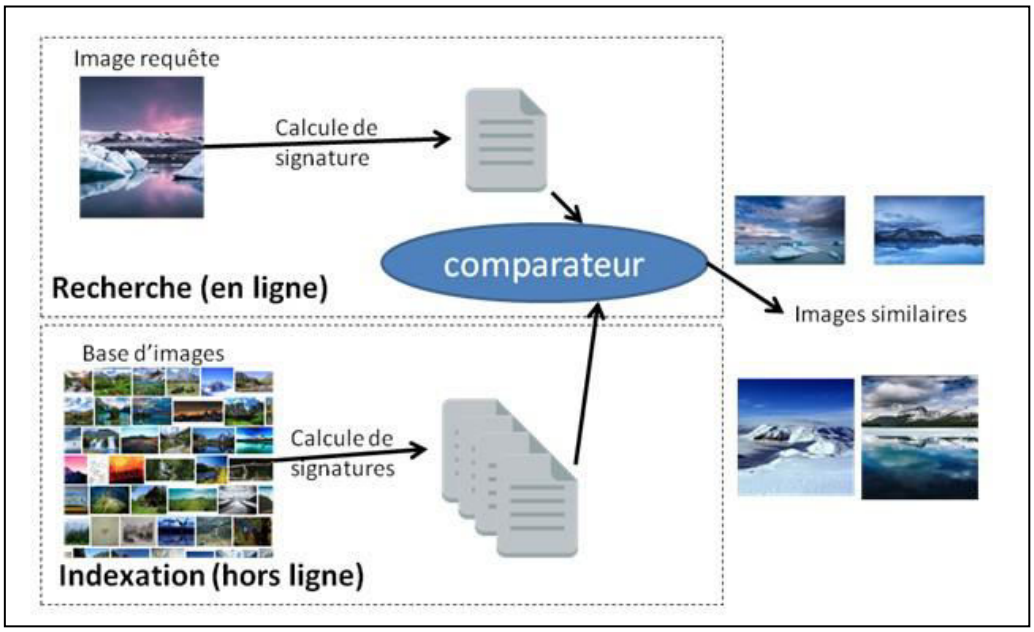


Fig.2. Principe général de la recherche d'images par le contenu.

I.4. L'indexation d'une vidéo

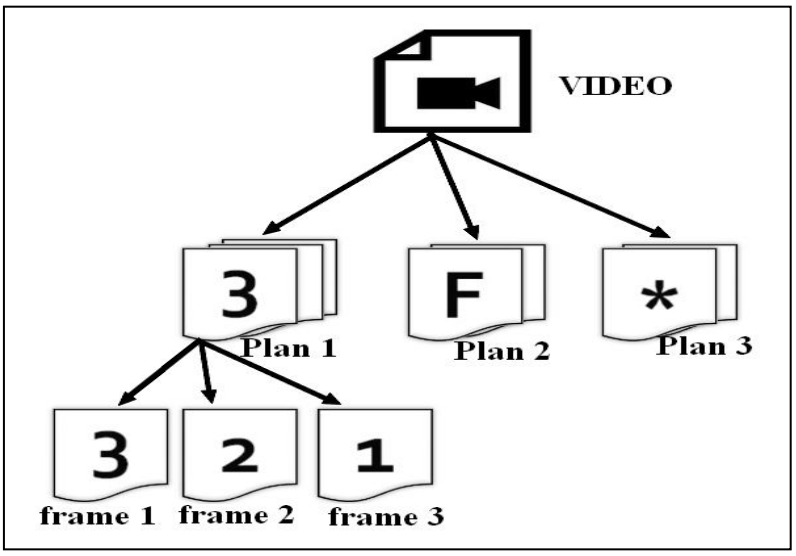


Fig.3.la segmentation d'une vidéo à des plans

Comme les données vidéo sont naturellement complexes; Donc une compréhension approfondie de ses caractéristiques uniques est essentielle pour comprendre les techniques qui la gèrent. Certaines caractéristiques importantes distinguent la vidéo d'autres classes de données. Premièrement, puisque la vidéo est stockée sous forme binaire; Par conséquent, contrairement aux données alphanumériques, la vidéo a une résolution plus élevée, un volume de données plus large, un ensemble plus grand de données qui peuvent être générées, une grande ambiguïté d'interprétation et nécessite davantage d'efforts d'interprétation.

Deuxièmement, la vidéo a une dimension spatiale et temporelle, alors que le texte n'est que statique non spatial et que l'image est spatiale statique. De plus, la sémantique vidéo est non structurée et contient généralement des relations complexes [13].

Les approches d'indexation vidéo peuvent être catégorisées en fonction des deux niveaux principaux de contenu vidéo: les caractéristiques de faible niveau (perceptuelles) et l'annotation de niveau élevé (sémantique).

Les principaux avantages des techniques d'indexation basées sur les caractéristiques bas sont :

- Ils peuvent être entièrement automatisés en utilisant des techniques d'extraction de caractéristiques, telles que l'analyse d'image et de son.
- Les utilisateurs peuvent utiliser la recherche de similarité à l'aide de certaines caractéristiques telles que la forme et la couleur des objets sur une image ou le volume de la piste sonore.

Le principal avantage de l'indexation sémantique à haut niveau est le soutien de plus des méthodes d'interrogation (requêtes) naturelles, puissantes et flexibles.

Par exemple, les utilisateurs peuvent parcourir une vidéo basée sur les concepts hiérarchiques sémantiques comme la classification topique et ils peuvent rechercher des vidéos particulières en fonction des mots-clés.

Cependant, ce type d'indexage est souvent réalisé grâce à une intervention manuelle puisque le processus de cartographie des fonctionnalités de bas niveau sur des concepts sémantiques n'est pas simple en raison des fossés sémantiques (semantic gaps).

L'annotation sémantique manuelle doit être minimisée car elle peut prendre beaucoup de temps, être partielle et incomplète.

I.4.1. L'indexation par la segmentation

Nous avons mentionné que l'indexation manuelle (intervention humaine –annotation-) est trop couteuse, dans cette partie, Nous saurons comment indexer une vidéo en se basent sur les frames qui la composent.

En général, l'indexation pourrait être effectuée sur le flux vidéo entier, mais il serait trop grossier. D'autre part, si l'indexation est basée sur chaque frame dans le clip, il serait trop dense car un frame ne contient souvent aucune information importante. Les chercheurs ont généralement indexé sur un groupe de frames séquentiels présentant des caractéristiques similaires [13], ce groupe de frames est appelé un plan.

Pour éviter la redondance de parcourir tous les frames du plan, le plan entier sera représenté par un seul frame puisque tous ces frames ont presque les mêmes caractéristiques, ce frame représentatif du plan est appelé le frame.

I.4.2. Le plan

Le plan est une séquence de frames (un ou plusieurs contigus) qui présente une action continue dans le temps et l'espace, par exemple dans la figure 4 les frames 1,2 et 3 appartient au même plan (porte le même sujet « chiffres »), les frames 4 et 5 un autre plan « lettres », les frame 6 et 7 «symboles».[12]

I.4.3. L'image clé

Ou bien *Keyframe*, c'est le frame représentatif du plan. Il est utilisé dans l'indexation c'est pour cela il est extrêmement important de le bien choisir.

On remarque que dans la figure 4 chaque plan est représenté par un keyframe, Dans la recherche ces keyframes seront comparés à l'image requête importé par l'utilisateur. [12]

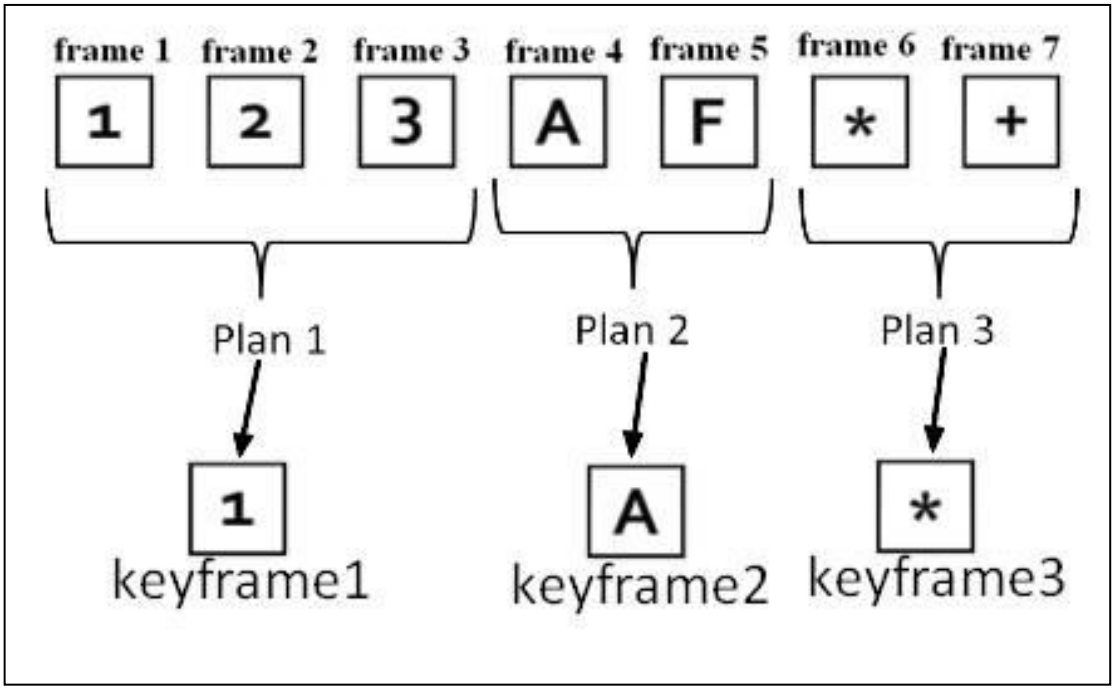


Fig.4. les plans et les images clés

I.5. Détection de coupure

Nous avons parlé de la segmentation de la vidéo pour obtenir les plans qui seront indexés par une seule image, la problématique est comment détecter le point de coupure (*cut detction*) ?

I.5.1. Le point de coupure

C'est le point où existe la séparation entre deux plans, il est généralement détecté par le calcul de la distance entre les frames contigus, quand la distance de similarité dépasse la valeur du seuil ça veut dire que les deux frames comparés sont maintenant des bords de nouveaux plans (figure.5).

I.5.2. Le seuil

C'est un paramètre essentiel et délicat qui sera comparé après chaque transition à la distance de similarité entre les deux frames contigus, la mauvaise sélection de la valeur du seuil peut facilement causer des problèmes durant le processus de découpage.

Il faut bien choisir la valeur du seuil car c'est elle qui déterminera le nombre et la longueur des plans, en d'autres termes le résultat de l'indexation est entièrement dépend au seuil.

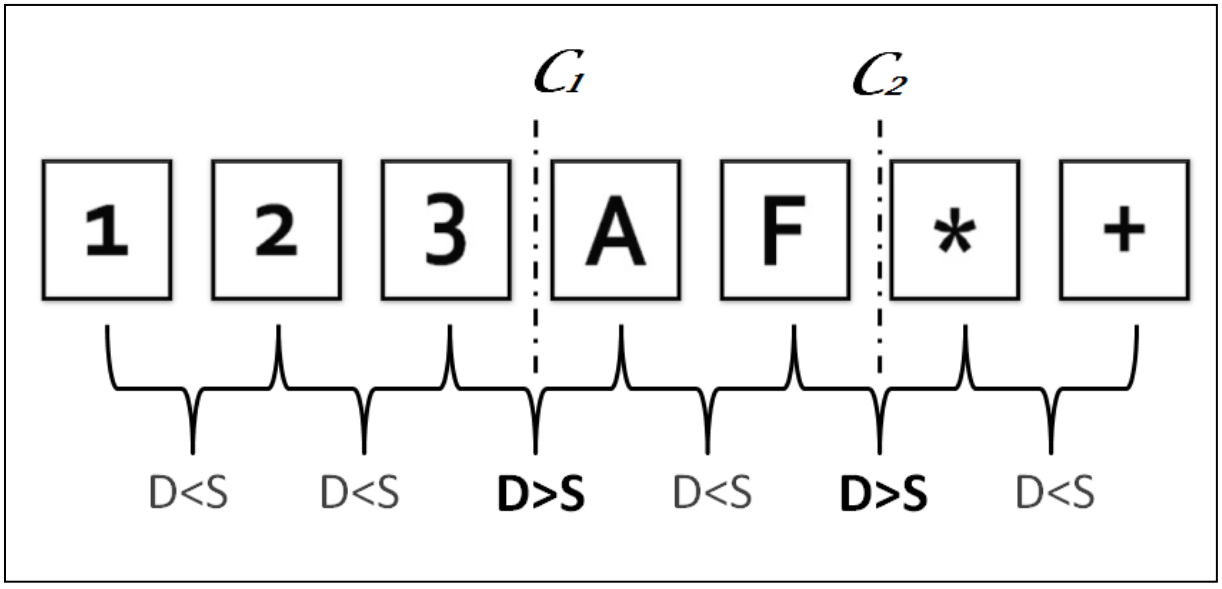


Fig.5. La détection d'un point de coupure.
D= distance de similarité ; S=seuil ; C=point de coupure.

I.5.3. Technique de coupure

En tant que technique de détection de coupure, une nouvelle technique basée sur l'Eq. (1) [3] a été développée, qui calcule l'intersection de l'histogramme de couleurs comme indiqué dans l'Eq. (2). Soit $h_{f,i}^{i}$ un rectangle d'histogramme de classe i au frame f. Puis l'histogramme normalisé bin $h_{f,i}$ est défini comme $h_{f,i}^{i} / \sum_j h_{f,i}^{i} (i,j = 1 \dots I : I = Q^3)$ avec Q est le nombre de classes R, G, B espace de couleur.

$$Cut(a) = min_b\, HI(h_a, h_b) - max_c(h_a, hc) \quad \dots\dots (1)$$

$$HI(h_a, h_b) = \sum_{i=1}^{I} min\left(h_{a,i}, h_{b,i}\right) \dots\dots\dots\dots\dots (2)$$

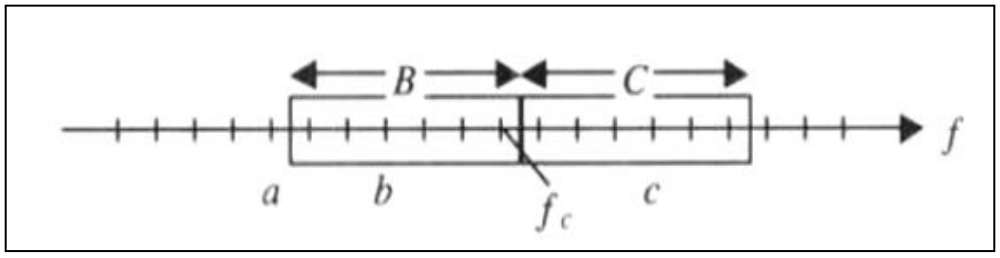


Fig 6. Extraction du point de coupure f_c

La figure 6 montre une configuration d'extraction de point de coupure. Deux fenêtres avec des frames B et C sont réglées sur le temps futur au cadre a. Lorsqu'une condition Cut(a)> θ_c> 0 est satisfaite, le point de coupe f_c est déterminé comme f_c = a + B, avec b = a + m (m = l ... B), c = a + B + n (n = 1 .. C).

On voit que cette méthode a une grande précision et le tableau suivant montre un résultat de l'extraction du point de coupure dans les conditions de Q= 8,8, θ_c=0,2, B,C= 6 [3].

Rappel = pertinents récupérés / tous pertinents

Précision = pertinents récupérés / tous récupérés

Correct	M	50
PAS de detection	D	3
Détection excessive	E	5
Rappel (%)	M/(M + D)	94.3%
Précision (%)	Ml(M + *E)*	90.9%

Table 7 :Résultat de l'extraction du point de coupure.

I.5.4. La coupure dure

La coupure dure « hard cut » c'est la transition principale, spécialement pour la vidéo non éditée et décrit le changement brusque entre deux plans, les histogrammes de couleur sont utilisés pour détecter la coupure dure comme l'algorithme de R. Lienhart, Sauf que la corrélation spatiale est prise en compte en divisant chaque frame en A = 3 zones horizontales sans être sensible au petit mouvement de la caméra et de l'objet. Pour la robustesse du bruit, le nombre de bins d'histogramme dans les espaces de couleur de HSV est réduit à 256 bins .La discontinuité D_a entre les frames est évaluée par la norme L_2 dans une fenêtre temporelle sur N = 5 frames. Un seuil adaptatif th_a (n) est calculé indépendamment pour chaque zone [2] :

$$th_a(n) = \alpha \cdot \left[\left(\sum_{m=n-N}^{n+N-1} D_a(m, m-1) \right) - D_a(n, n-1) \right] + \beta$$

Les constantes régulent les seuils adaptatifs, dans les essais empiriques α = 2,5 et β = 8,7 conduisent au meilleur résultat. Une coupe dure est détectée, si D_a (n, n - 1)> th_a (n) et Da (n + 1, n) <th_a (n + 1) est vrai pour toutes les zones (a = 1..3).

I.5.5. Dissolution

Une dissolution est définie par un chevauchement temporel de quelques frames du plan disparaissant et apparaissant [2]. La variance de la luminance pendant le chevauchement

progresse sur une parabole avec un minimum au centre de la transition (Fig. 3). les candidats sont extraits par les caractéristiques de la première et la deuxième dérivées de $Y\sigma^2$. En raison de la faible qualité d'image et des opérations de caméra rapides, cette approche produit de nombreux candidats à la dissolution. Nos efforts reposent sur la vérification de ces candidats. Les extrema de la dérivée seconde marquent le point de départ et le point final d'un candidat de dissolution qui est utilisé pour générer une dissolution idéalement modélisée.

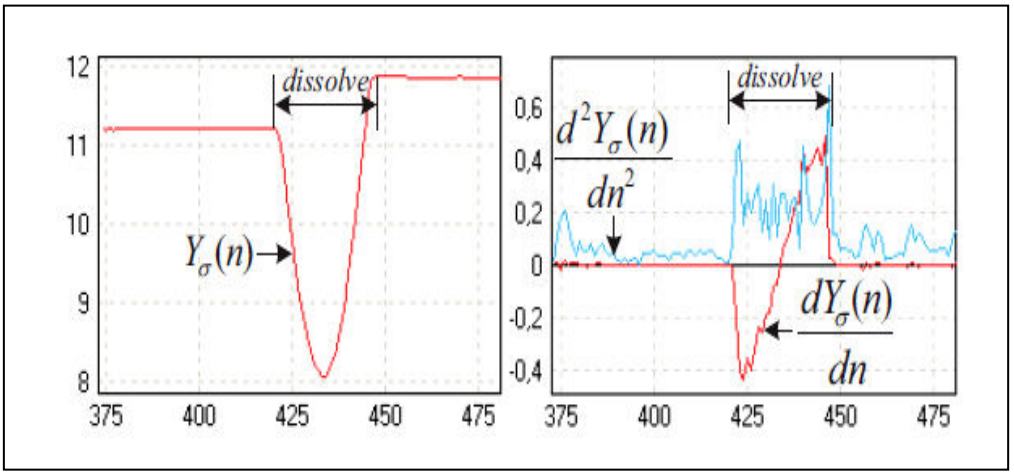


Fig.8. Courbe de déviation standard (gauche), première et deuxième dérivées de la courbe de déviation standard (droite).

La vérification se fait en considérant les aspects suivants. La corrélation croisée entre le déroulement des premières dérivées de la région candidate et la dissolution idéale doit dépasser le seuil $th_{cc} = 0,9$. De plus, la moyenne de la première dérivée doit dépasser un seuil positif faible.

I.6. Conclusion

Dans ce chapitre nous avons vu la donné vidéo, l'indexation et la recherche par le contenu et comment segmenter une vidéo en keyframes qui seront les éléments fondamentaux de l'indexation vidéo. La seule chose qui reste abstruse est comment la comparaison entre deux images numériques est possible.

Chapitre II : L'extraction des caractéristiques visuelles

II.1. Introduction

Il est trop difficile de faire des traitements sur le contenu brut d'une vidéo ou bien d'un frame mais il faut d'abord extraire les caractéristiques de ce frame, et pour faire cela on a besoin d'un descripteur. Il y'a plusieurs type de descripteurs et chacun a ses propre spécificités et ses points forts.

L'extraction des descripteurs est réalisée sur les frames composants de la vidéo, pour des raisons multiples (l'indexation, la recherche, la comparaison…). Qu'est ce qu'un descripteur d'une image ?

II.2. Les descripteurs

Le descripteur, la signature ou l'extraction de caractéristiques visuelles est de transformer l'image a des données manipulables avec des opérations et des formules mathématiques calculées sur les pixels d'une image numérique afin d'utilisées ces données ultérieurement pour autres traitements telles que la détection d'objets ou la recherche d'images par le contenu. [16]

On peut catégoriser les descripteurs selon leurs types à :

- Descripteurs de couleur
- Descripteurs de texture
- Descripteurs de forme
- Descripteurs de points d'intérêts

II.2.1. Le niveau de gris

Le niveau de gris est la valeur de l'intensité lumineuse en un point. La couleur du pixel peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux intermédiaires. Donc pour représenter les images à niveaux de gris, on peut attribuer à chaque pixel de l'image une valeur correspondant à la quantité de lumière renvoyée. Cette valeur peut être comprise par exemple entre 0 et 255.[14] [15].

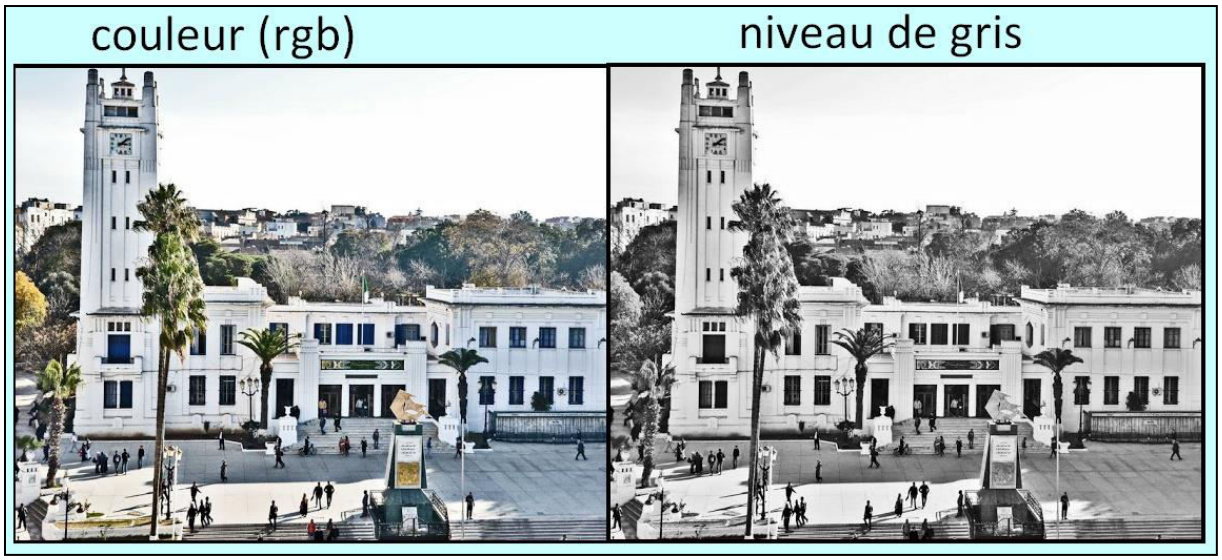


Fig.9. image coloré et en niveau de gris

II.2.2. Descripteurs de couleur

C'est le plus souvent réalisé à partir de l'histogramme de couleur. Les variantes proviennent du choix de l'espace couleur et du nombre de bins par composantes selon l'espace de couleur choisie. [17]

- ***Le modèle RGB***

Est un model digital représentant les 3 couleurs primaires additive (Red, Green, Blue / Rouge, Vert, Bleu) C'est l'un des espaces de couleurs perceptuel et uniforme qui ressemble approximativement la façon de perception des couleurs pour l'humain. [14]

II.2.3. Descripteurs de texture

La texture de l'image est apparue comme une primitive visuelle importante pour rechercher et parcourir de grandes collections de modèles de recherche similaires. Une image peut être considérée comme une mosaïque de textures et ces caractéristiques associées aux régions peuvent être utilisées pour indexer les images.[18]

La texture de l'image est utile dans la navigation, la recherche et la récupération d'images.

- Gabor Filter bank

La représentation de Gabor s'est révélée optimale dans le sens de minimiser l'incertitude articulaire bidimensionnelle dans l'espace et la fréquence.

Ces filtres peuvent être considérés comme des détecteurs de bord et de ligne accordable à une échelle, et les statistiques de ces caractéristiques microscopique peuvent être utilisées pour caractériser la texture sous-jacente.

II.2.4. Descripteurs de forme

Le descripteur de forme est un vecteur d'un certain nombre de paramètres dérivés d'un calcul qui repose sur la synthèse de valeurs de pixels dans une image numérique contenant la silhouette d'un objet donné. [21] C'est très robuste au changement d'éclairage.[22]

II.2.5. Descripteurs de points d'intérêts

Les Descripteur de points d'intérêts est un descripteur qui est largement invariant à l'échelle, la rotation, le bruit de l'image et de petits changements de point de vue [10], il extrait des points comme suit:

a. ***Détection des espaces extrêmes dans l'espace:*** la première étape du calcul est de mettre en œuvre efficacement en utilisant une fonction de différence de Gaussien pour identifier les points d'intérêt potentiels qui sont invariants à l'échelle et à l'orientation.

b. ***Localisation des points d'intérêt:*** à chaque emplacement candidat, un modèle détaillé est adapté pour déterminer l'emplacement et l'échelle. Les points clés (d'intérêt) sont sélectionnés en fonction des mesures de leur stabilité.

c. ***Affectation d'orientation:*** Une ou plusieurs orientations sont attribuées à chaque emplacement de point central en fonction des directions locales du dégradé d'image. Toutes les opérations futures sont effectuées sur des données d'image qui ont été transformées par rapport à l'orientation, l'échelle et l'emplacement assignés pour chaque caractéristique, ce qui assure l'invariance de ces transformations.

d. ***Descripteur de points d'intérêt:*** les gradients d'image locaux sont mesurés à l'échelle sélectionnée pour chaque région autour de chaque point clé (keypoint). Ceux-ci sont transformés en une représentation qui permet des niveaux significatifs de distorsion de forme locale et de changement d'illumination. (figure 10)

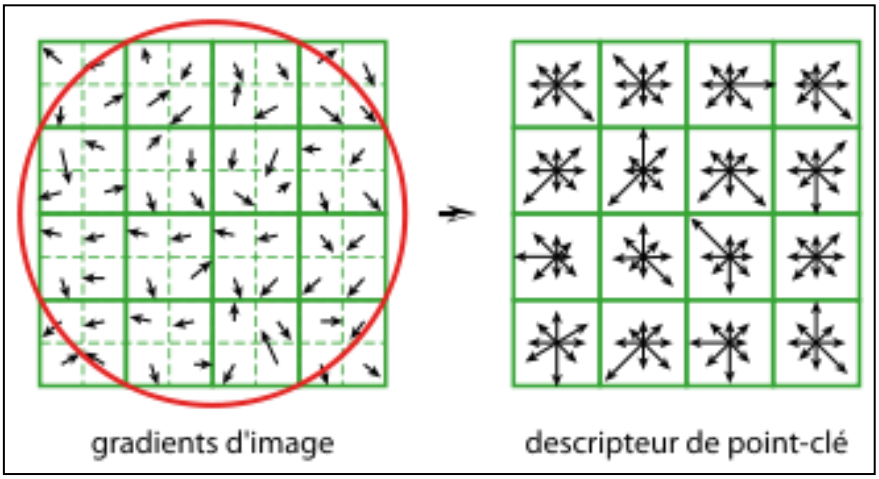


Fig.10. la transformation des gradients à des descripteurs de points d'intérêt

II.3. Histogramme de couleurs

Les algorithmes d'élaboration d'histogramme caractérisent une image par sa distribution de couleur ou son histogramme [9]. Un histogramme n'est rien d'autre qu'un graphe qui représente toutes les couleurs et le niveau de leur occurrence dans une image quel que soit le type de l'image. Peu de propriétés de base sur une image peuvent être obtenues à partir d'un histogramme. Il peut être utilisé pour définir un seuil pour le criblage des images. La forme et la concentration des couleurs dans l'histogramme seront les mêmes pour des objets similaires, même si elles sont de couleurs différentes. L'identification des objets dans une image en échelle de gris est la plus facile car l'histogramme est presque similaire car les objets ont les mêmes couleurs pour les mêmes objets [9].

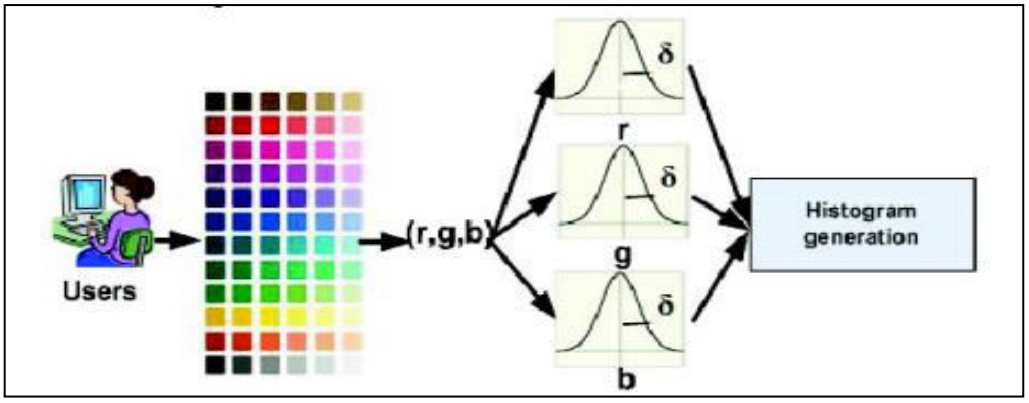


Fig.11. Histogramme de couleur

En général, toute image contient des informations utiles et indésirables. Le système doit différencier les deux. Imaginant une image où une personne lisant un livre est l'information utile et l'arrière-plan, les gens et le marché sont les données indésirables. Le système doit regrouper le motif répété pour identifier les objets dans l'image.

II.3.1. Mesures de similarité de l'histogramme

Une image peut être représentée par un histogramme de couleur, défini par un schéma de quantification de couleur appliqué à un modèle de couleur [6]. Afin d'exprimer la similitude de deux histogrammes dans un actif numérique, on utilise une distance métrique, on trouve une grande variété de mesures de distance entre histogrammes (exemple : euclidienne, intersection ...). En mesurant la distance entre les images composantes une séquence de frames, on peut marquer les distances qui sont largement différentes aux autres déjà calculées et comme ça on peut diviser la séquence de frames a des sous-séquences qui seront considérées comme des plans.

II.4. SIFT (Transformation de caractéristiques visuelles invariante à l'échelle)

SIFT (Scale Invariant Feature Transform) est une approche pour détecter et extraire des descripteurs de caractéristiques locales qui sont raisonnablement invariantes aux changements d'éclairage, de mise à l'échelle, de rotation, de bruit d'image et de petits changements de point de vue. [11]

SIFT détecte des points d'intérêt dans l'image en utilisant l'opérateur Différence de Gauss (DOG). Les points sont sélectionnés en tant qu'extrémités locales de la fonction DoG. A chaque point d'intérêt, un vecteur de caractéristique est extrait. Sur un certain nombre d'échelles et sur un voisinage autour du point d'intérêt, l'orientation locale de l'image est estimée en utilisant les propriétés d'image locales pour fournir l'invariance contre la rotation. Ensuite, un descripteur est calculé pour chaque point détecté, sur la base des informations locales de l'image.

Localiser les points d'intérêt est l'étape clé, est également la première étape dans la reconnaissance d'objet en utilisant la méthode SIFT est de générer les points caractéristiques stables. La figure 12 donne un processus complet sur la façon de trouver et de décrire les points caractéristiques SIFT.

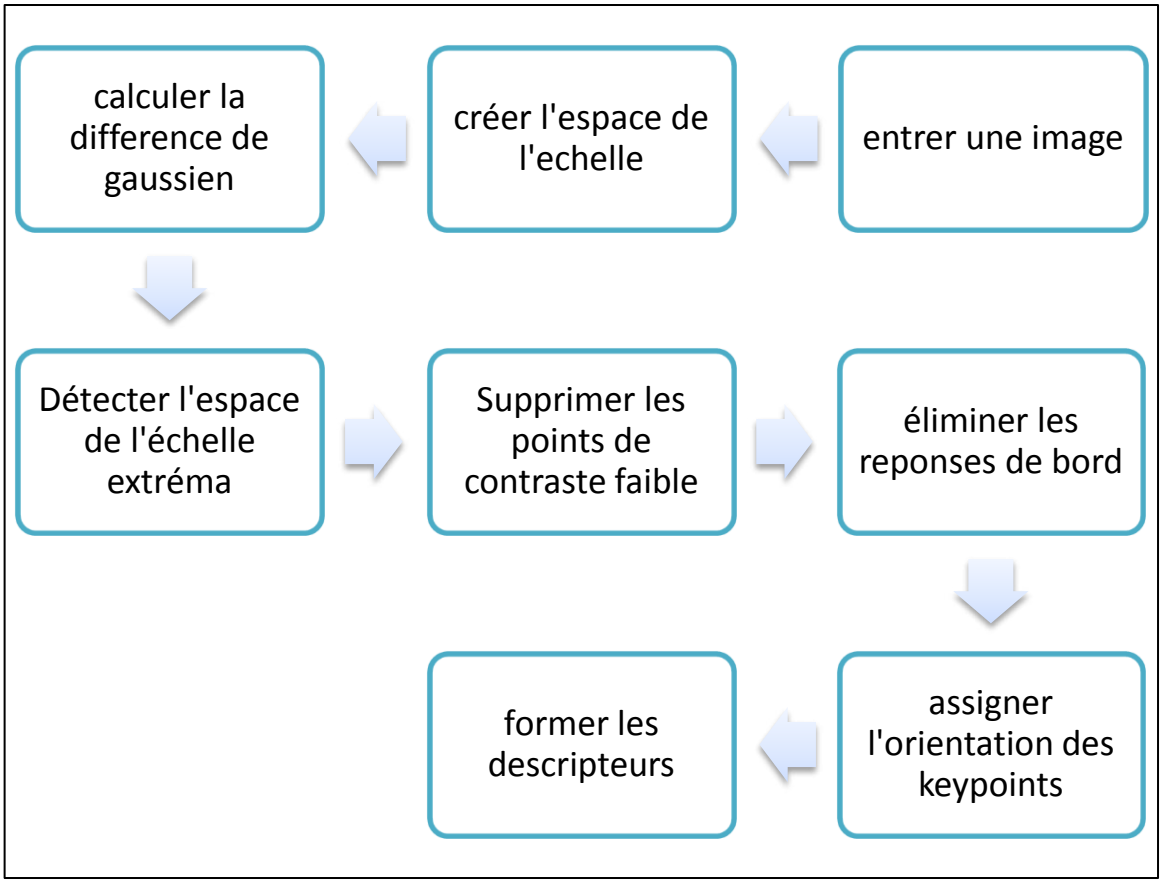


Fig 12. Diagramme de localisation des points d'intérêt

II.4.1. Les points forts de SIFT

a. Les caractéristiques SIFT sont toutes des caractéristiques naturelles des images. Ils sont invariant favorablement à la traduction d'image, à l'échelle, à la rotation, à l'illumination, au point de vue, au bruit etc.

b. Bonne spécialité, riche en informations, adapté à une correspondance rapide et précise dans une masse de base de données de caractéristiques.

c. Vitesse relativement rapide. La vitesse de SIFT peut même satisfaire le processus en temps réel après que l'algorithme SIFT est optimisé. [11]

II.5. SURF (caractéristiques robustes accélérées)

La bonne performance des SIFT par rapport aux autres descripteurs est remarquable. Son mélange d'informations localisées de façon grossière et la distribution des caractéristiques liées au gradient semblent donner un bon pouvoir distinctif tout en évitant les effets des erreurs de localisation en termes d'échelle ou d'espace. L'utilisation des forces et orientations relatives des gradients réduit l'effet des changements photométriques.

Le descripteur proposé de SURF (Speeded Up Robust Features) [1] est basé sur des propriétés similaires, avec une complexité dépouillée encore plus. La première étape consiste

à fixer une orientation reproductible sur la base d'informations provenant d'une région circulaire autour du point d'intérêt. Ensuite, la construction d'une région carrée alignée sur l'orientation sélectionnée, et en extrayant le descripteur SURF. Ces deux étapes sont maintenant expliquées à tour de rôle. En outre, une version verticale du descripteur (U-SURF) qui n'est pas invariant à la rotation d'image est proposée et donc ce dernier est plus rapide à calculer et mieux adapté pour des applications où la caméra reste plus ou moins horizontale.

II.5.1. Affectation d'orientation

Afin d'être invariant à la rotation, nous identifions une orientation reproductible pour les points d'intérêt. Pour cela, nous calculons d'abord les réponses d'ondelettes de Haar dans les directions *x* et *y*, représentées sur la Fig. 13, et cela dans un voisinage circulaire de rayon 6*s* autour du point d'intérêt, avec *s* l'échelle à laquelle le point d'intérêt a été détecté. L'étape d'échantillonnage est également dépendante de l'échelle et choisie pour être *s* En accord avec le reste, les réponses en ondelettes sont également calculées à cette échelle de courant *s*. En conséquence, à des échelles élevées, la taille des ondelettes est grande. Par conséquent, nous utilisons à nouveau des images intégrales pour un filtrage rapide. Seules six opérations sont nécessaires pour calculer la réponse en direction *x* ou *y* à n'importe quelle échelle. La longueur latérale des ondelettes est de 4*s*. [1]

Une fois les réponses en ondelettes calculées et pondérées avec un Gaussien ($\sigma = 2{,}5s$) centré au point d'intérêt, les réponses sont représentées comme vecteurs dans un espace avec la force de réponse horizontale à côté de l'abscisse et la force de réponse verticale à côté de l'ordonnée. L'orientation dominante est estimée en calculant la somme de toutes les réponses dans une fenêtre d'orientation coulissante couvrant un angle de $\frac{\pi}{3}$. Les réponses horizontales et verticales dans la fenêtre sont additionnées. Les deux réponses sommées donnent alors un nouveau vecteur. Le plus long de ces vecteurs prête son orientation au point d'intérêt. La taille de la fenêtre coulissante est un paramètre qui a été choisi expérimentalement. Les petites tailles tirent sur les réponses à ondelettes dominantes simples. Les deux entraînent une orientation instable de la région d'intérêt. Notez que l'U-SURF ignore cette étape.

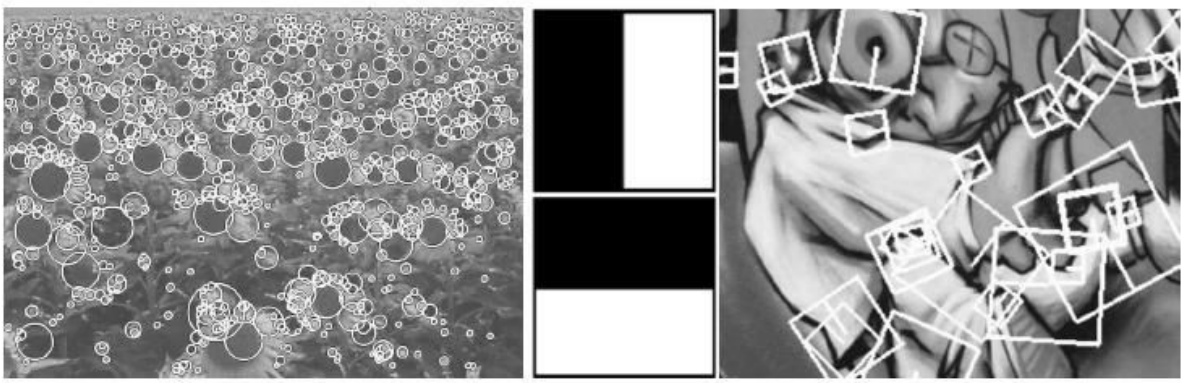


Fig. 13. Différents index SURF

La figure 13 représente :

Gauche: Points d'intérêt détectés pour un champ de tournesol, ce genre de scènes montre clairement la nature des caractéristiques des détecteurs à base de Hesse.

Milieu: types d'ondelettes Haar utilisés pour SURF.

Droite: Détail de la scène Graffiti montrant la taille de la fenêtre du descripteur à différentes échelles.

Pour l'extraction du descripteur, la première étape consiste à construire une zone carrée centrée autour du point d'intérêt et orientée selon l'orientation sélectionnée dans la section précédente. Pour la version verticale, cette transformation n'est pas nécessaire. La taille de cette fenêtre est de 20s. Des exemples de telles régions carrées sont illustrés sur la Fig. 13.

II.5.2. Indexation rapide pour le matching

Pour l'indexage rapide pendant l'étape du matching, le signe du Laplacien (c'est-à-dire la trace de la matrice Hessienne) pour le point d'intérêt sous-jacent est inclus. Typiquement, les points d'intérêt se trouvent dans des structures de type blob. Le signe du Laplacien distingue les taches lumineuses sur les fonds sombres de la situation inverse. Cette fonctionnalité est disponible sans coût de calcul supplémentaire car elle a déjà été calculée pendant la phase de détection. Dans la phase d'adaptation, nous ne comparons les caractéristiques que si elles ont le même type de contraste, voir la figure 14. Par conséquent, cette information minimale permet une correspondance plus rapide [4], sans réduire la performance du descripteur. Notez que cela est également avantageux pour des méthodes d'indexation plus avancées. Par exemple. Pour les arbres k-d (kd trees), cette information supplémentaire désigne un hyperplan significatif pour diviser les données, par opposition au choix aléatoire d'un élément ou à l'aide de statistiques de caractéristiques.

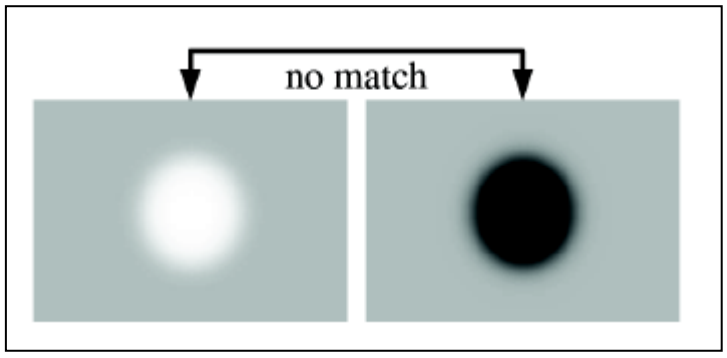


Fig.14.le matching entre deux point d'intérêt

La figure 14 illustre : si le contraste entre deux points d'intérêt est différent (sombre sur fond clair contre lumière sur fond sombre), le candidat n'est pas considéré comme un match valable.

II.6. Histogrammes de gradients orientés (pour la détection d'humain)

II.6.1. Un Aperçu

La méthode est basée sur l'évaluation d'histogrammes locaux bien normalisés d'orientations de gradient dans une grille dense. Des manières similaires sont progressivement utilisées au cours de la dernière décennie. L'idée de base est que l'apparence et la forme de l'objet local peuvent souvent être assez bien caractérisés par la distribution des gradients locaux d'intensité ou des directions de bord, même sans connaissance précise des positions de gradient ou de bord qui correspond. En pratique, ceci est réalisé en divisant la fenêtre d'image en petites régions spatiales ("cellules"), Pour chaque cellule accumulant un histogramme local de directions de gradient ou d'orientations de bords sur les pixels de la cellule. Les entrées d'histogramme combinées forment la représentation. Pour une meilleure invariance à l'illumination, à l'ombre, etc., il est également utile de normaliser les réponses locales avant de les utiliser. Cela peut être fait en accumulant une mesure de l'histogramme local «énergie» sur des régions spatiales un peu plus grandes («blocs») et en utilisant les résultats pour normaliser toutes les cellules dans le bloc. On référence aux blocs de descripteurs normalisés comme Histogramme de gradient orienté (HOG ***H**istogram of **O**riented **G**radient*) des descripteurs. Carreler la fenêtre de détection avec une grille dense (en fait, chevauchante) de descripteurs HOG et l'utilisation du vecteur caractéristique combiné dans un classificateur de fenêtre SVM conventionnel donne notre chaîne de détection d'humain (fig 15). [20]

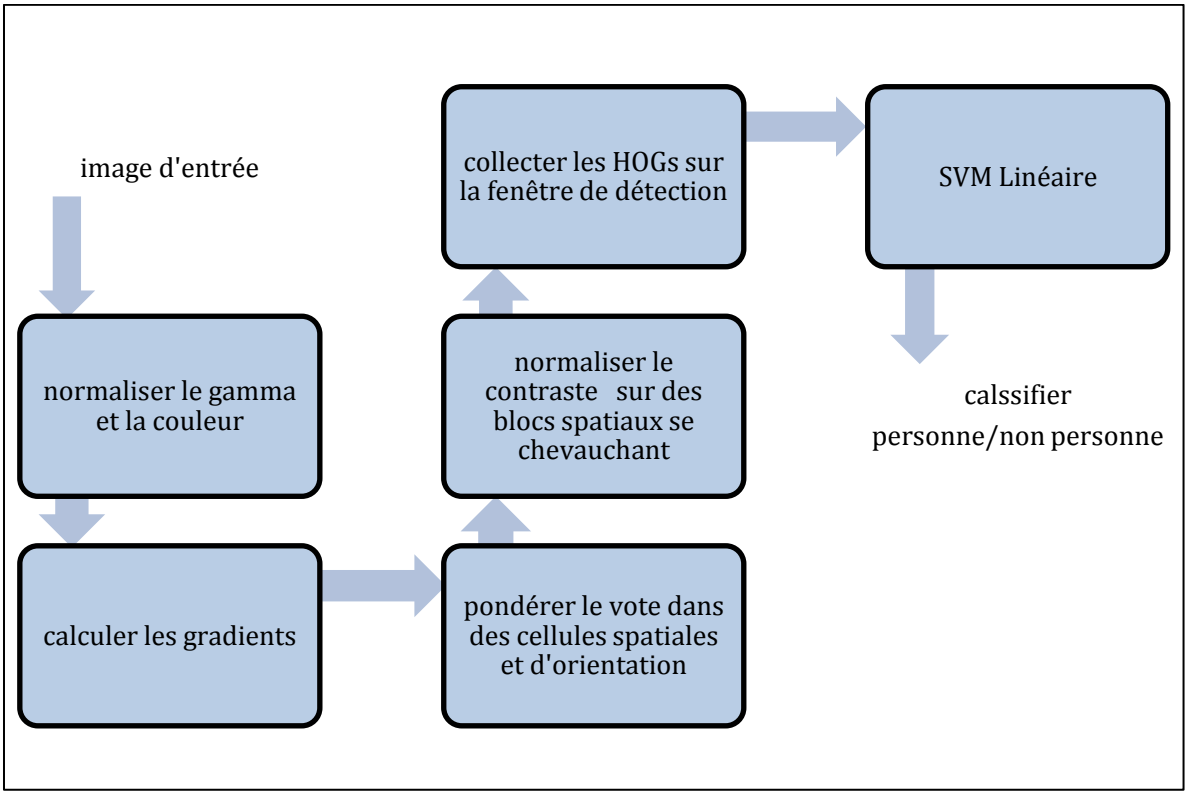


Fig 15. Un aperçu de la chaîne d'extraction de caractéristiques et de détection d'objets.

La fenêtre du détecteur est carrelée avec une grille de blocs superposés dans laquelle les vecteurs d'histogramme de gradient orienté sont extraits. Les vecteurs combinés sont introduits dans une SVM linéaire pour la classification objet / non-objet. La fenêtre de détection est balayée sur l'image à toutes les positions et toutes les échelles, et la suppression classique non maximale est exécutée sur la pyramide de sortie pour détecter les instances d'objet. [20]

II.6.2. Étude de mise en œuvre et de rendement

On donne maintenant des détails sur les implémentations HOG et on étudie systématiquement les effets des différents choix sur la performance des décideurs. On va référer les résultats au détecteur par défaut qui présente les propriétés suivantes, décrites ci-dessous: Espace couleur RVB sans correction gamma; [-1, 0, 1] filtre gradient sans lissage; Gradient linéaire votant en 9 cases d'orientation en 0 ◦-180◦; 16 × 16 blocs de pixels de quatre cellules de 8 × 8 pixels; Fenêtre spatiale gaussienne avec σ = 8 pixels; Normalisation des blocs L2-Hys (Lowe-style clipped L2 norm); Espace de 8 pixels (donc 4 fois la couverture de chaque cellule); 64 × 128 fenêtre de détection; Linéaire SVM.

II.6.3. Normalisation des couleurs/Gamma

Après l'évaluation de plusieurs représentations de pixel, comprenant le niveau de gris et les espaces de couleur RGB et LAB optionnellement avec gamma, ces normalisations n'ont qu'un effet modeste sur la performance, peut-être parce que la normalisation ultérieure du descripteur atteint des résultats similaires. Nous utilisons l'information de couleur quand elle est disponible. Les espaces de couleur RVB et LAB donnent des résultats comparables, mais la restriction aux niveaux de gris réduit la performance de 1,5% à 10^{-4} FPPW (False Positives Per Window/faux positif par fenêtre). La compression racine carrée de gamma de chaque canal de couleur améliore les performances à faible FPPW (de 1% à 10^{-4} FPPW), mais la compression LOG est trop forte et l'aggrave de 2% à 10^{-4} FPPW.

II.6.4. Calcul du gradient

La performance du détecteur est sensible à la façon dont les gradients sont calculés, mais le schéma le plus simple s'avère être le meilleur. Un teste des gradients calculés en utilisant un lissage gaussien suivi de plusieurs masques dérivés et discrets. Plusieurs échelles de lissage ont été testées incluant σ = 0 (aucune). Les masques testés comprenaient divers dérivés 1-D (non concentrés [-1, 1], centré [-1, 0, 1] et cubiques corrigés [1, -8, 0, 8, -1] ainsi que 3 × 3 masques de Sobel et 2 × 2 diagonales $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$, $\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$ (les masques dérivés 2-D centraux les plus compacts). Les masques simples 1-D [-1, 0, 1] à σ = 0 fonctionnent le mieux. L'utilisation de masques plus grands semble toujours diminuer la performance et le

lissage l'endommage significativement, pour les dérivées gaussiennes, passer de $\sigma = 0$ à $\sigma = 2$ réduit le taux de rappel de 89% à 80% à 10^{-4} FPPW. A $\sigma = 0$, les filtres à largeur 1-D corrigée cubique sont environ 1% plus mauvais que [-1, 0, 1] à 10^{-4} FPPW, tandis que les 2×2 masques diagonaux sont 1,5% plus mauvais. L'utilisation de masques dérivés non concentrés [-1, 1] diminue également la performance (de 1,5% à 10^{-4} FPPW), probablement parce que l'estimation d'orientation souffre du fait que les filtres x et y sont basés sur des centres différents.

Pour les images en couleurs, le calcul se fait sur des gradients distincts pour chaque canal de couleur et celui avec la plus grande norme comme vecteur de gradient du pixel est pris.

II.6.5. Orientation spatial du binning

L'étape suivante est la non-linéarité fondamentale du descripteur. Chaque pixel calcule un vote pondéré pour un canal d'histogramme d'orientation de bord basé sur l'orientation de l'élément de gradient centré sur celui-ci et les votes sont accumulés dans des cases d'orientation sur des régions spatiales locales que nous appelons des cellules. Les cellules peuvent être rectangulaires ou radiales (secteurs log-polaires). Les cases d'orientation sont uniformément espacées sur 0 ◦ - 180◦ (gradient "non signé") ou 0 ◦-360◦ (gradient "signé").

Pour réduire l'aliasing, les votes sont interpolés bilinéairement entre les centres voisins des bins dans l'orientation et la position. Le vote est une fonction de l'amplitude du gradient au pixel, soit l'amplitude elle-même, son carré, sa racine carrée, soit une forme découpée de l'amplitude représentant la présence / absence faible d'un bord au pixel. En pratique, l'utilisation de l'amplitude elle-même donne les meilleurs résultats. Prenant la racine carrée réduit les performances légèrement, tandis que l'utilisation de la présence de bord binaire la baisse de manière significative (de 5% à 10^{-4} FPPW).

II.7. Conclusion

Dans ce chapitre nous avons cité quelque méthodes de L'extraction de caractéristiques visuelles comme l'histogramme de couleur, SIFT, SURF et HOG. Nous avons aussi défini que ce que c'est un descripteur, ses types et son rôle majeur pour la comparaison entre les images numériques.

Chaque descripteur a des avantages qui ne sont pas couverts par les autres descripteurs.

Chapitre III : Indexation et segmentation par SIFT

III.1. Introduction

Le coût de calcul du traitement vidéo est très élevé et la détection de changement du plan est l'une des raisons de ce coût. La détermination de la portée du plan est un élément essentiel dans la plupart des applications de traitement vidéo, en particulier pour l'indexage vidéo et la récupération vidéo basée sur le contenu.

Dans ce chapitre nous présentons notre approche de la recherche de point de coupure et son emplacement afin de pouvoir indexer la vidéo et éviter plein de calcul dans l'étape de la recherche.

III.2. Notre approche

L'idée générale de notre approche est de générer des images clés et/ou des descripteurs d'images qui représentent la globalité de la vidéo, ces derniers peuvent être utilisés pour la recherche ou pour autre fonctionnalités ultérieurement.

Notre algorithme se base sur le découpage de la vidéo à des plans qui sont en suite figurés par un seul frame a fin d'éviter le parcours de toute la vidéo. Nous pouvons décrire l'algorithme en trois étapes principales (figure 16).

a) ***Prétraitement vidéo***

Cette étape n'est pas applicable pour les flux vidéo en temps réel, elle consiste à parcourir le contenu des vidéos enregistrées sur le disque dur et extraire tous les frames qui composent la vidéo choisie.
Dans le prétraitement les frames extraits de la vidéo sont converties à des images de niveau de gris pour qu'elles soient traitées par les descripteurs *SIFT* d'une manière facile.

b) ***Calcul d'index***

- *Pour la vidéo enregistrée*

On calcule le SIFT du premier frame et le SIFT du frame qui suit, on calcule on suite la différence entre les SIFTs obtenus en appliquant une distance de similarité si la cette dernière surpasse la valeur du seuil donc on obtient un nouveau point de coupure (changement de plans), sinon on reste dans le même plan.

- *Pour le flux vidéo en temps réel*

La capture des frames et leurs conversions au niveau de gris se fait au même temps mais le processus est le même comme la vidéo enregistrée.

Après la détection d'un point de coupure nous obtenons un plan qui contient des frames qui éventuellement ont les mêmes spécificités, qui seront représentés par un seul frame (keyframe) qui représente au mieux notre plan, pendant ce temps le processus continue jusqu'à la fin de la vidéo ou bien de son arrêt.

c) *Sélection et enregistrement des keyframes*

Au début le keyframe est le premier frame du plan, mais après on a une option de sélectionner à nouveau un autre keyframe qui représente au mieux le plan.

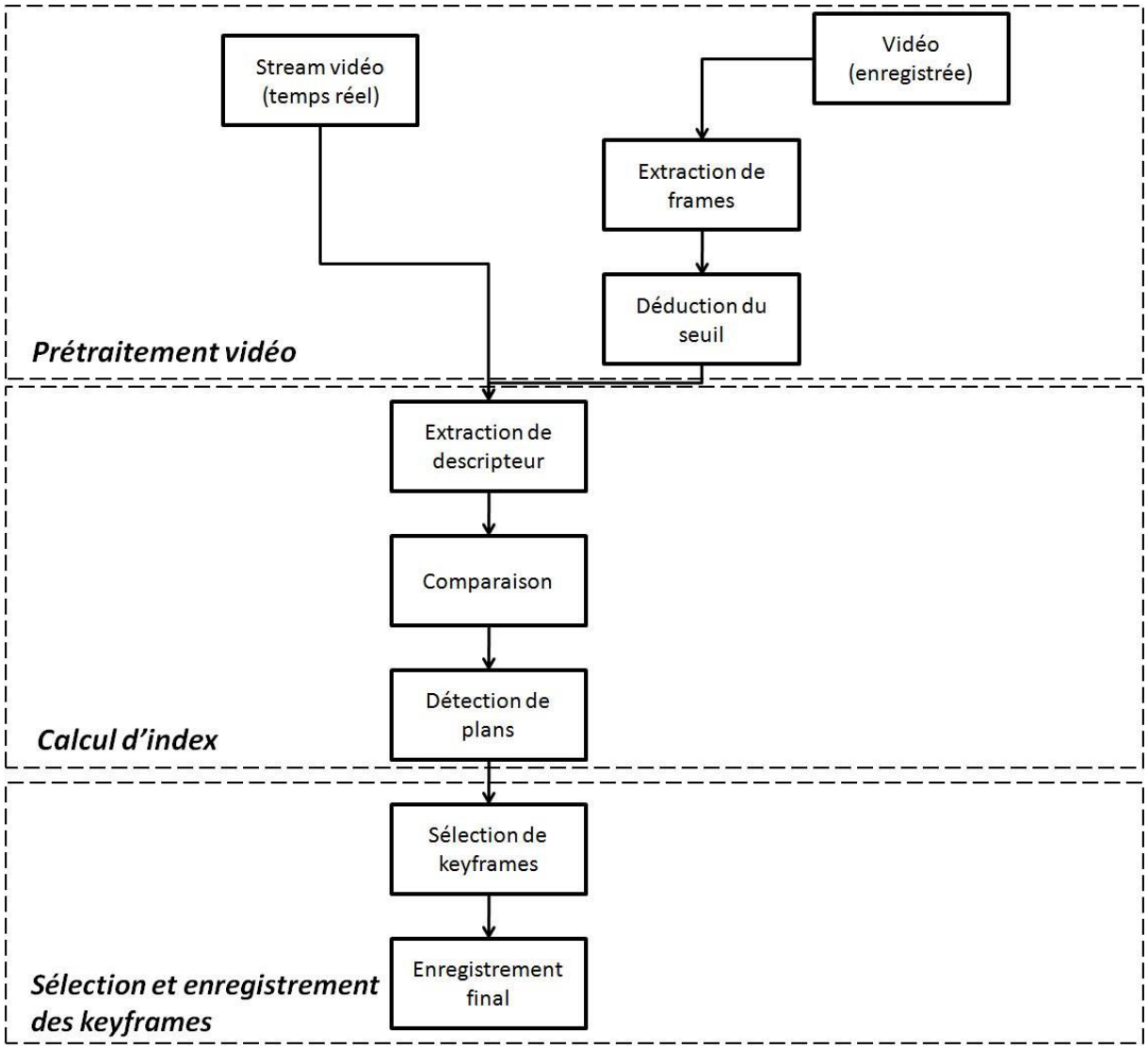


Fig.16.représentation générale de l'algorithme

III.3. Pseudo-algorithme

FC = null

Entrée : *fichier vidéo/flux vidéo en temps réel* ***comme*** *V*

Sortie : *les keyframes* ***comme*** *FC*

FC = $frame_1$

Tans que *(frame(V) n'est pas vide)*

I_1 ← $frame_i$

I_2 ← $frame_{i+1}$

S_1 ← Sift(I_1)

S_2 ← Sift(I_2)

Si *(la_déférence(S_1,S_2) > Seuil)*

FC.ajouter(I_2)

Fin SI

i ← i + 1

Fin tans que

III.4. Extraction de frames

Cette étape se fait dans le cas du découpage de la vidéo enregistrée à des frames (images simples), les frames sont convertis à des images de niveau de gris.

Pourquoi travailler avec le niveau de gris ?

Les valeurs que prend une image (à une seule valeur) sont habituellement des intensités puisqu'elles sont un enregistrement de l'intensité du signal sur le capteur, par exemple, le nombre de photons ou l'amplitude d'une fonction d'onde mesurée. [5]

L'intensité est une quantité positive. Si l'image est représentée visuellement à l'aide de nuances de gris (comme une photographie en noir et blanc), les valeurs de pixels sont appelées niveaux de gris. Bien sûr, d'une manière générale, une image peut prendre des

valeurs multiples à chaque pixel (comme une image en couleur « exemple de RGB »), ou une image peut avoir des valeurs de pixels négatives, auquel cas il ne s'agit pas d'une fonction d'intensité. Dans tous les cas, les valeurs d'image doivent être quantifiées pour le traitement numérique.

La quantification est le processus de conversion d'une image à valeur continue qui a une portée continue (ensemble de valeurs qu'il peut prendre) dans une image à valeur discrète qui a une plage discrète. Cela se fait ordinairement par un processus d'arrondissement, qui facilite le traitement de l'image et cela est très efficace pour la détection des limites et la reconnaissance. [5]

III.4.1. Le seuil

Nous avons mentionné que la valeur du seuil est extrêmement importante puisque c'est elle qui sera la condition de découpage de l'ensemble des frames à des plans. Ces derniers vont être indexés par une seule image (le keyframe), de tout ça on comprend qu'il faut bien choisir la valeur du seuil.

La déduction automatique de la valeur du seuil est faite après l'extraction de tous les frames, en suite une sélection aléatoire de frames se fait, qui est basée sur le nombre total de frames de la vidéo.

Les frames aléatoires qui sont extraits subissent une extraction de caractéristique « points d'intérêt SIFT » le nombre de point avec la taille de la vidéo vont définir la valeur du seuil, tel que la moyenne des points d'intérêt est calculée est divisée sur la largeur de la vidéo.

Dans certaines vidéos la valeur automatique du seuil peut donner des résultats indésirables, pour cette raison la définition manuelle de la valeur du seuil est largement conseillée.

III.4.2. Détection de changement de plan

Le changement du plan est la où il faut détecter le point de coupure. Cela est le but quintessentiel de notre approche et pour le réaliser il faut d'abord extraire les caractéristiques visuelles pour effectuer une comparaison entre les frames.

III.5. L'extraction des caractéristiques visuelles SIFTs

Nous avons travaillé avec les SIFT en raison de ça capacité énorme de l'invariance à l'échelle, à la rotation, à l'illumination, au point de vue et au bruit qui permet d'obtenir des résultats assez bons dans ces différentes situations.

Pour la vidéo déjà enregistrée, l'extraction des descripteurs SIFTs est faite directement sur les frames déjà recueilli est préparés dans l'étape prétraitement, par contre dans le flux vidéo, l'extraction des descripteurs SIFT est réalisée juste après la capture du frame et la préparation en grayscale.

III.6. La comparaison de frames

Il s'agit de comparer les descripteurs extraits de deux frames qui se suivent, et cette comparaison n'est qu'un calcul de distance entre chaque descripteur de point d'intérêt SIFT (keypoint) en utilisant une distance mathématique.

Chaque keypoint contient un descripteur de gradients sommés, les frames qui ont plus de descripteur de keypoint presque identique sont similaires qui veut dire qu'ils appartiennent au même plan, et qui ne partagent rien comme descripteur de point d'intérêt similaire donnent un point de coupure, cette procédure s'appelle le *matching*.

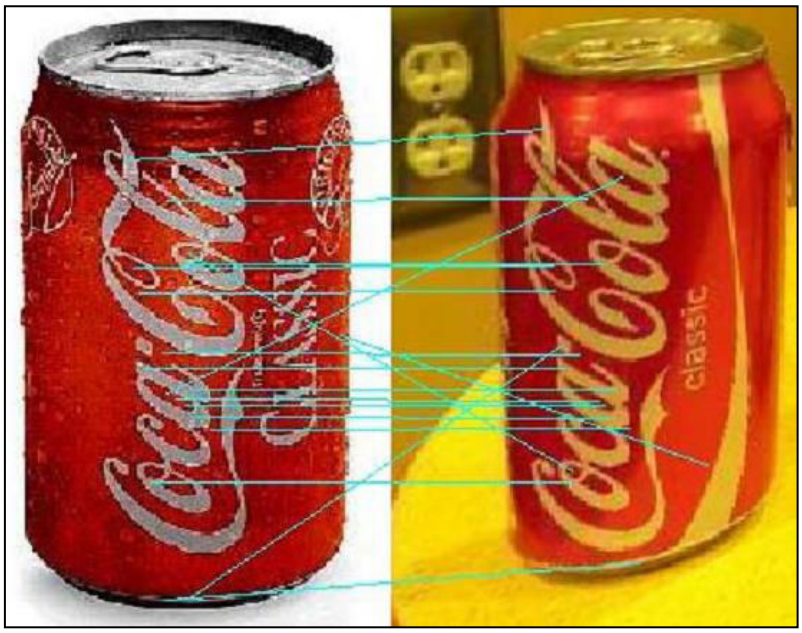


Fig.17. le matching entre deux images avec SIFT

III.6.1. Calcule de distance

Pour comparer les descripteurs des points d'intérêt et établir le matching, il faut calculer la différence entre les vecteurs de gradients sommés avec une distance mathématique parmi plusieurs nous détaillons quelque unes :

- ***Distance de Manhattan***

La distance entre deux points dans une grille est basée sur un chemin strictement horizontal et / ou vertical par opposition à la diagonale. La distance de Manhattan est la somme simple des composantes horizontale et verticale. Elle est appelée aussi taxi-distance ou L1.[8]

$$d(A,B) = |X_A - X_B| + |Y_A - Y_B|$$

- ***Distance euclidienne***

La distance euclidienne entre deux points est la longueur du segment de ligne qui les relie. Appelée la distance L2. [8]

$$d(A,B) = \sqrt{(X_A - X_B)^2 + (Y_A - Y_B)^2}$$

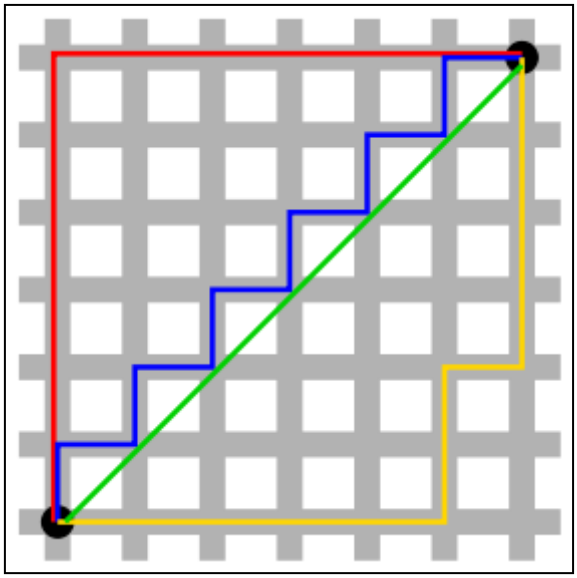


Fig.18. Distance de Manhattan (chemins rouge, jaune et bleu)
contre distance euclidienne en vert.

- ***Distance euclidienne carrée***

La distance Euclidienne standard peut être quadrillée afin de placer un poids progressivement plus important sur les objets qui sont plus éloignés. Dans ce cas, l'équation devient

$$d(A,B) = \sqrt{(X_A - X_B)^2 + (Y_A - Y_B)^2}^2$$

La distance euclidienne au carré 'L2sqr'[8] n'est pas une métrique, car elle ne satisfait pas l'inégalité triangulaire, mais elle est fréquemment utilisée dans les problèmes d'optimisation dans lesquels les distances doivent être comparées.

III.7. Détection des plans

Nous avons dit que la détection de plans estle fait de segmenter la vidéo à des ensembles de frames qui partagent les mêmes caractéristiques et après avoir comment comparer entre deux signatures SIFTs entre eux ça sera facile de séparer l'ensemble de frames à des plans indépendants qui seront le noyau de l'indexation.

A ce niveau, le mieux est de définir pour chaque plan une image que le représente, cette image est à son tour indexée par le SIFT et l'ensemble de ces index permet de représenter le contenu de la vidéo.

La sélection des keyframes est détaillé dans le paragraphe qui suit.

III.8. La sélection des keyframes

Dans notre approche les keyframes seront les premiers frames de chaque plan puisque nous avons dit que les frames du même plan représentent le même contenu si la segmentation était bonne donc la sélection d'un frame quelconque d'un plan nous donne généralement les mêmes résultats.

Dans l'indexation, l'optimisation maximale est toujours requise donc la sélection du frame qui donne beaucoup d'information sur le plan serait donc extrêmement efficace et utile, et notre approche fournit un outil qui permet de générer à nouveau des keyframes qui ont une tendance à porter plein d'information par rapport aux autres frames.

L'outil qui nous fournissons de la sélection automatique de keyframes est de comparer les points d'intérêts SIFT de chaque frame aux autres du même plan, le frame qui contient le plus de points d'intérêts serait le nouveau frame représentatif du plan (figure 19), cela nous donne plus de descripteur de point d'intérêts à comparer avec les descripteurs qui seront extraits de l'image requête généré par l'utilisateur lors de la recherche.

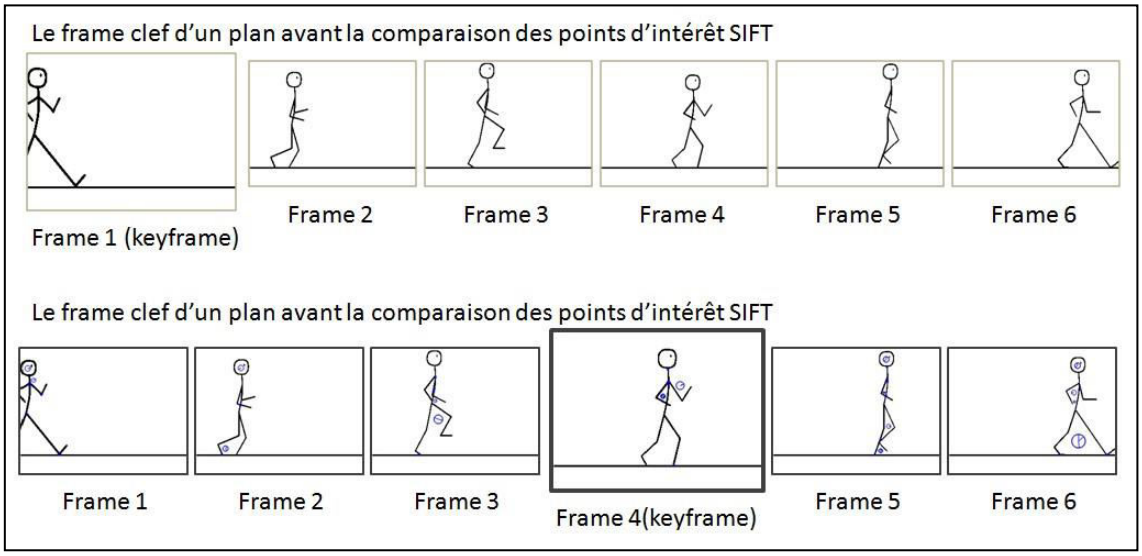


Fig.19. la réélection des keyframes.

Apres la sélection des keyframes, Nous savons bien que maintenant nous avons les frames qui servent comme résumé de la vidéo à indexer, dans ce stade nous avons pu extraire les informations essentielles du flux vidéo, ce travail nous permet d'éviter la comparaison de la totalité de la vidéo mais de comparer que les images qui sont importantes.

III.8.1. L'enregistrement des descripteurs SIFT

L'enregistrement des descripteurs de keypoints SIFT est le but de notre travail, ces SIFT sont des données d'une matrice de dimension N*128 (n= nombre de keypoints), les descripteurs enregistrés sont les index de la vidéo, chaque descripteur SIFT représente les carectéristique d'un frame sous forme numérique facilement comparable.

Nous avons ajouté une option d'enregistrer les keyframes sous forme d'images visuelles pour donner plus de liberté dans la recherche, par exemple si nos résultats images seraient utilisé avec une autre manière de recherche, il suffit d'extraire les caractéristiques visuelles avec des nouveaux type d'outils d'extraction de caractéristiques.

III.9. Conclusion

Ce chapitre présente la méthode utilisée dans notre approche et donne une vue générale de son déroulement et ses trois étapes principales, et il donne une idée sur la comparaison entre les frames en utilisant le descripteur de point d'intérêt SIFTs et l'utilisation des keypoints SIFT pour déduire un frame qui représente au mieux le plan.

Chapitre IV : Implémentation

IV.1. Introduction

Le chapitre précédent donne une idée générale sur le fonctionnement de notre approche, et il décrit les techniques utilisées pour obtenir les points de coupures, et comment extraire les frames importants pour l'indexation finale.

Dans ce chapitre nous nous intéresserons à la mise en œuvre de notre approche, et nous montrons des résultats après un test.

IV.2. Environnement matériel et logiciel

IV.2.1. Ressources utilisées

- Les ressources physiques utilisées sont :
- Processeur pentium® dual-core T4200 d'une fréquence de 2.0GHZ.
- Une mémoire vive d'une capacité de 2GO.
- Une carte graphique de 732 MB.

Et pour ce qui est du coté soft :

- Système d'exploitation : Windows 7.
- Langage de programmation : Visual Basic .NET

IV.2.2. Le Langage de programmation

Visual Basic .NET (VB.NET) est un langage de programmation multi-paradigme orienté objet, implémenté sur .NET Framework. Microsoft a lancé VB.NET en 2002 comme successeur de son langage Visual Basic d'origine. Bien que la partie ".NET" du nom a été supprimée en 2005, cet article utilise "Visual Basic [.NET]" pour se référer à toutes les versions de Visual Basic versions depuis 2002, afin de distinguer entre elles et Visual Basic classique. Avec Visual C #, c'est l'une des deux principales langues ciblant le framework .NET.

L'environnement de développement intégré (IDE) de Microsoft pour développer en langage Visual Basic .NET est Visual Studio. La plupart des éditions Visual Studio sont commerciales; Les seules exceptions sont Visual Studio Express et Visual Studio Community,

qui sont des logiciels gratuits. En outre, .NET Framework SDK comprend un compilateur de ligne de commande freeware appelé vbc.exe. Mono comprend également un compilateur VB.NET de ligne de commande.

IV.3. Fenêtres du prototype

On décrit dans cette section le contenu et le rôle de chaque fenêtre.

IV.3.1. Fenêtre CutDet

Cette fenêtre est la fenetre initiale de notre programme , elle contient beaucoup d'options qui seront expliquées par la suite.

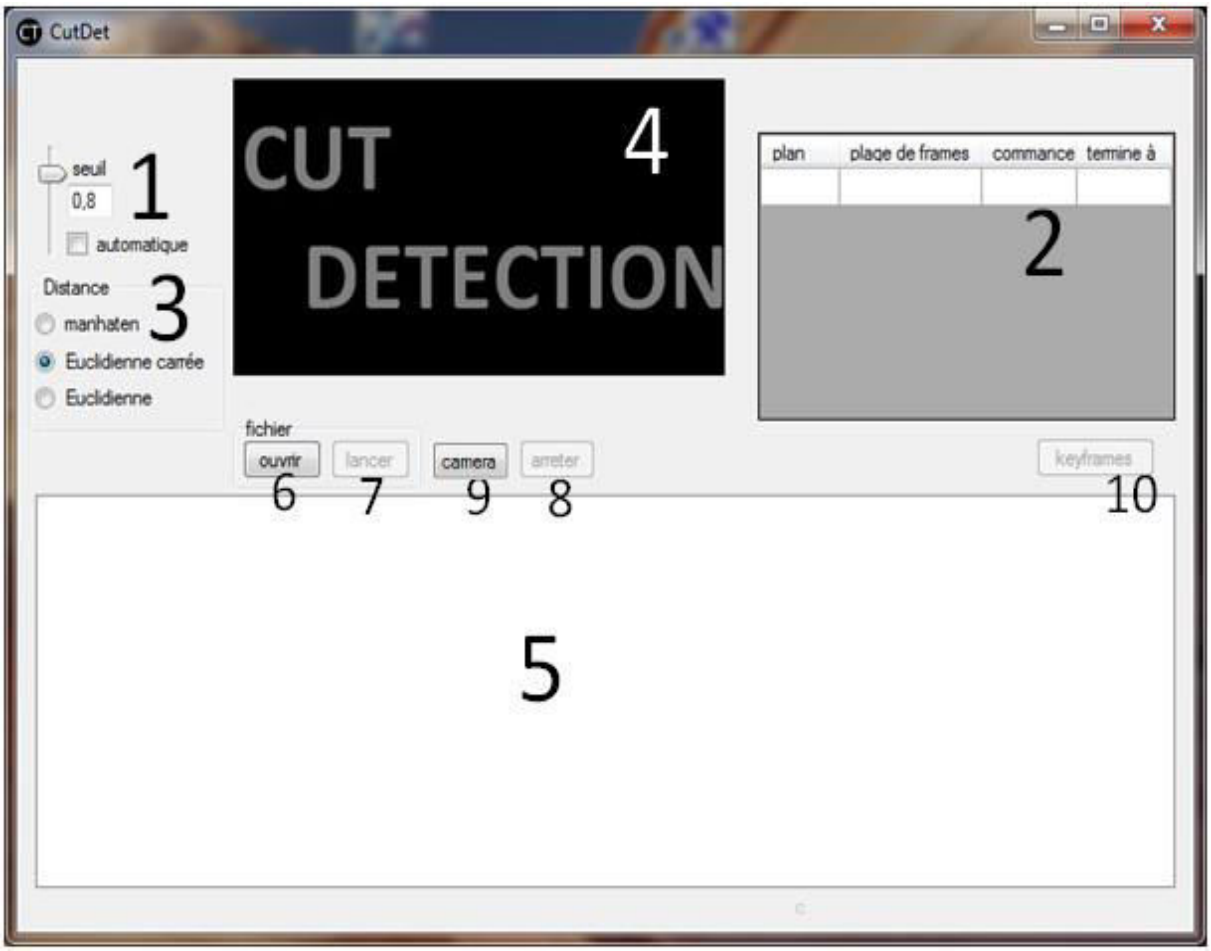


Fig.20. la fenêtre principale du prototype

1: Le seuil

Le figure 21 montre l'interface qui permet déterminer le seuil de la détection de coupure à la fenêtre principale CutDet.

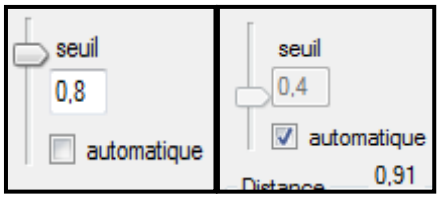


Fig.21. le seuil

Gauche : seuil manuel, la valeur est de 0,8. par exemple

Droite : seuil automatique, la valeur est de 0 ,91.

2: Affichage des plans

plan	plage de frames	commance	termine
1	1 - 1	0;0	0;0
2	2 - 4	0;0	0;0
3	5 - 11	0;0	0;1
4	12 - 12	0;1	0;1
5	13 - 14	0;1	0;1
6	15 - 16	0;1	0;1

Fig.22. Affichage des plans

Ce tableau est une sorte de datagrid qui se remplira durant l'exécution, il contient quatre colonnes :

- **Plan** : qui définit le numéro du plan.
- **Plage de frames** : affiche le numéro (classement) du premier et le dernier frame qui composent le plan.
- **Commence** : le temps où le plan commence .
- **Termine** : le temps où le plan termine.

3: choix de la distance

Trois distances possible a choisir (Manhattan, euclidienne simple et carrée), ils sont toutes expliquées dans le paragraphe III.6.

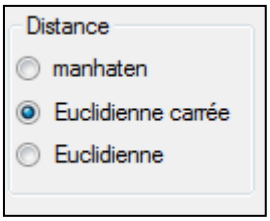


Fig.23. choix de la distance

4: l'affichage du frame en cours de traitement

C'est une sorte de picturebox qui affiche le frame la ou le traitement se fait.

Fig.24. l'affichage du frame en cours de traitement

5: Affichage des frames de plans

C'est une sorte de listeview qui affiche touts les frames parcourus et séparer entre les plan au cas de l'existence d'un point de coupure.

Fig.25. Affichage des frames de plans

6: Le bouton ouvrir

En cliquant sur ce bouton on aura une fenêtre de dialogue qui nous permet de choisir une vidéo à traiter.

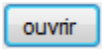


Fig.26. Le bouton ouvrir

7: Le bouton lancer

Le bouton permet de commencer l'opération de découpage .

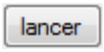


Fig.27. Le bouton lancer

8: Le bouton arrêter

Pour arrêter l'exécution du traitement.

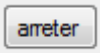


Fig.28. Le bouton arrêter

1: Le bouton caméra

En cliquant sur le bouton caméra nous pourrions traiter le flux vidéo provenant de la caméra relié avec notre pc.

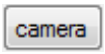


Fig.29.Le bouton caméra

1: Le bouton keyframes

Ce bouton affiche une nouvelle fenetre qui nous donne les keyframes choisis apres l'échantillonnage de la vidéo.

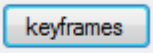


Fig.30.Le bouton keyframes

IV.3.2. Fenêtre Keyframes

C'est la fenêtre qui affiche les keyframes de tous les plans extrais, elle se compose d'un listeview la où s'affichent les keyframes et cinq bouton qui sont :

Régénérer les keyframes : pour sélectionner à nouveaux les keyframes selon le nombre des point d'intérêt (chapitreIII.8).

Enregistrer les SIFTs : enregistrer le descripteur SIFT de chaque keyframe sous forme d'un document texte Txt.

Enregistrer les keyframes : enregistrer les keyframes sous forme d'image JPG.

Ouvrir le dossier : pour ouvrir le dossier ou l'enregistrement a été fait, chaque bouton ouvre le dossier de l'action faite par le bouton qui est juste à coté.

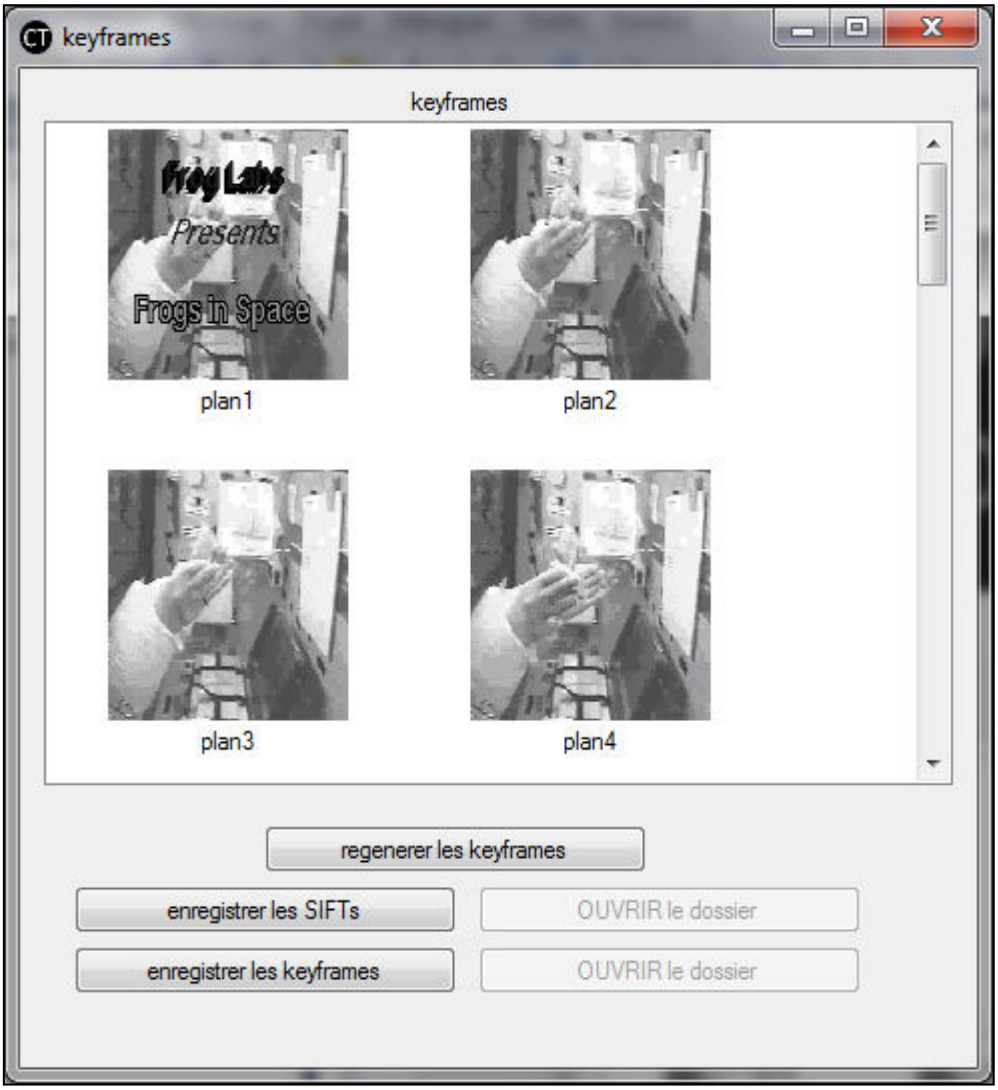


Fig.31. la fenêtre keyframes

IV.4. Exemple d'expérimentation

Pour éclaircir le principe de notre approche, nous donnons dans ce qui suit deux exemples d'illustration, les résultats sont retournés par notre prototype.

IV.4.1. Le premier exemple

Notre exemple est une vidéo enregistrée de cette propriété :

- Taille (Size) : 325 x 288
- Longueur (Length) : 22 secondes
- FPS : 25,00

IV.4.1.a. L'étape d'indexation

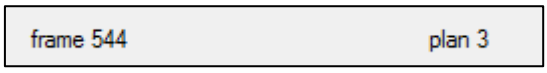


Fig.32.le résultat de la première expérimentation

Les résultats de cette étape sont 544 frames qui sont distribués en 3 plans, des informations supplémentaires sur la figure 33.

plan	plage de frames	commance	termine à
1	1 - 25	0;0	0;1
2	26 - 506	0;1	0;20
3	507 - 544	0;20	0;22

Fig.33. informations sur les plans de la première expérimentation

La figure 33 décortique le résultat comme suit :

- **Le premier plan** commence du premier frame jusqu'au vingt-cinquième frame, et de la seconde 0 à la seconde 1.

- **Le deuxième plan** commence du vingt sixième frame jusqu'au cinq cent sixième frame, et de la seconde 1 à la seconde 20.

- **Le troisième plan** commence du cinq cent septième frame jusqu'au cinq cent quarante-quatrième frame, et de la seconde 20 à la seconde 22.

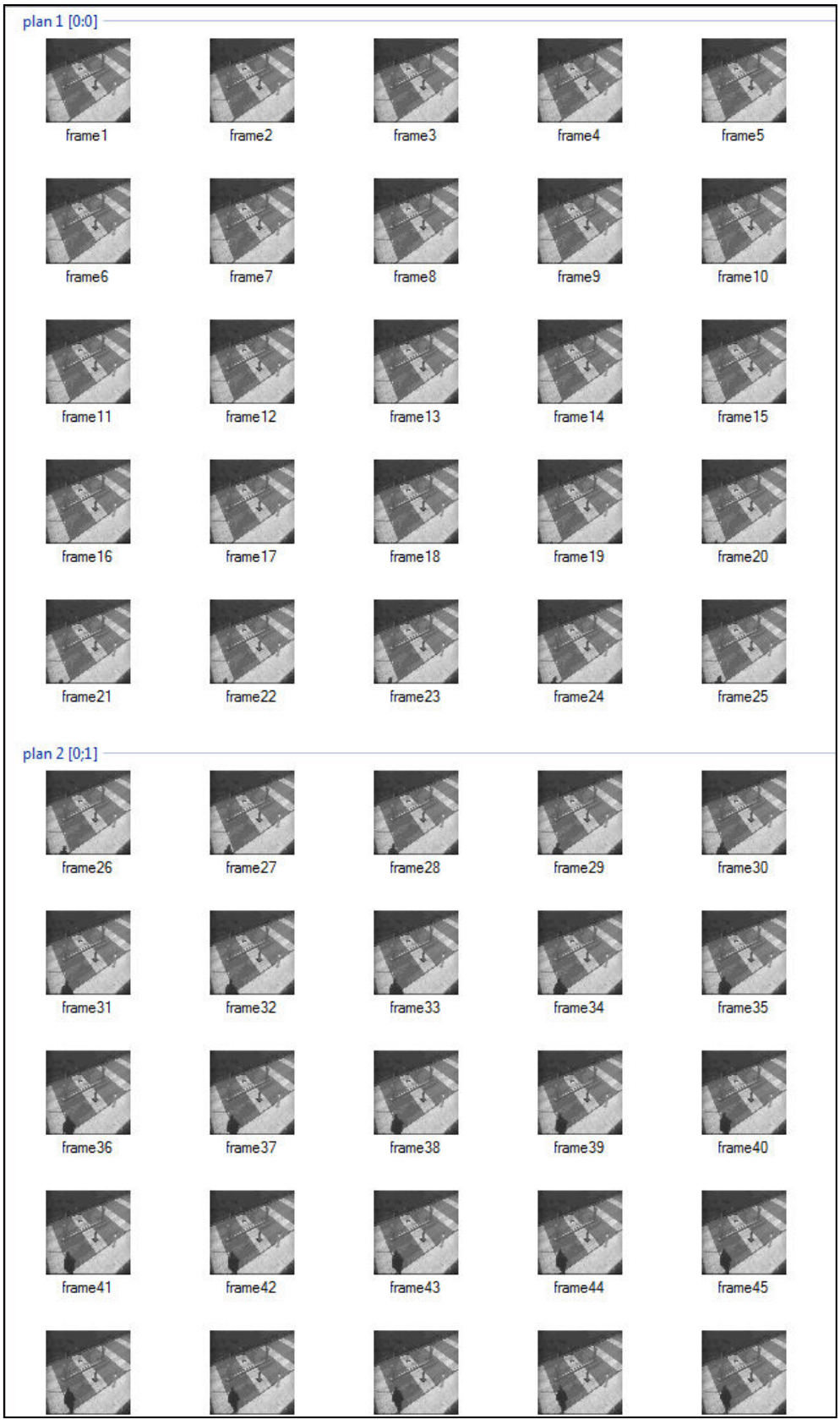


Fig.34. les frames et les plans (1-45)

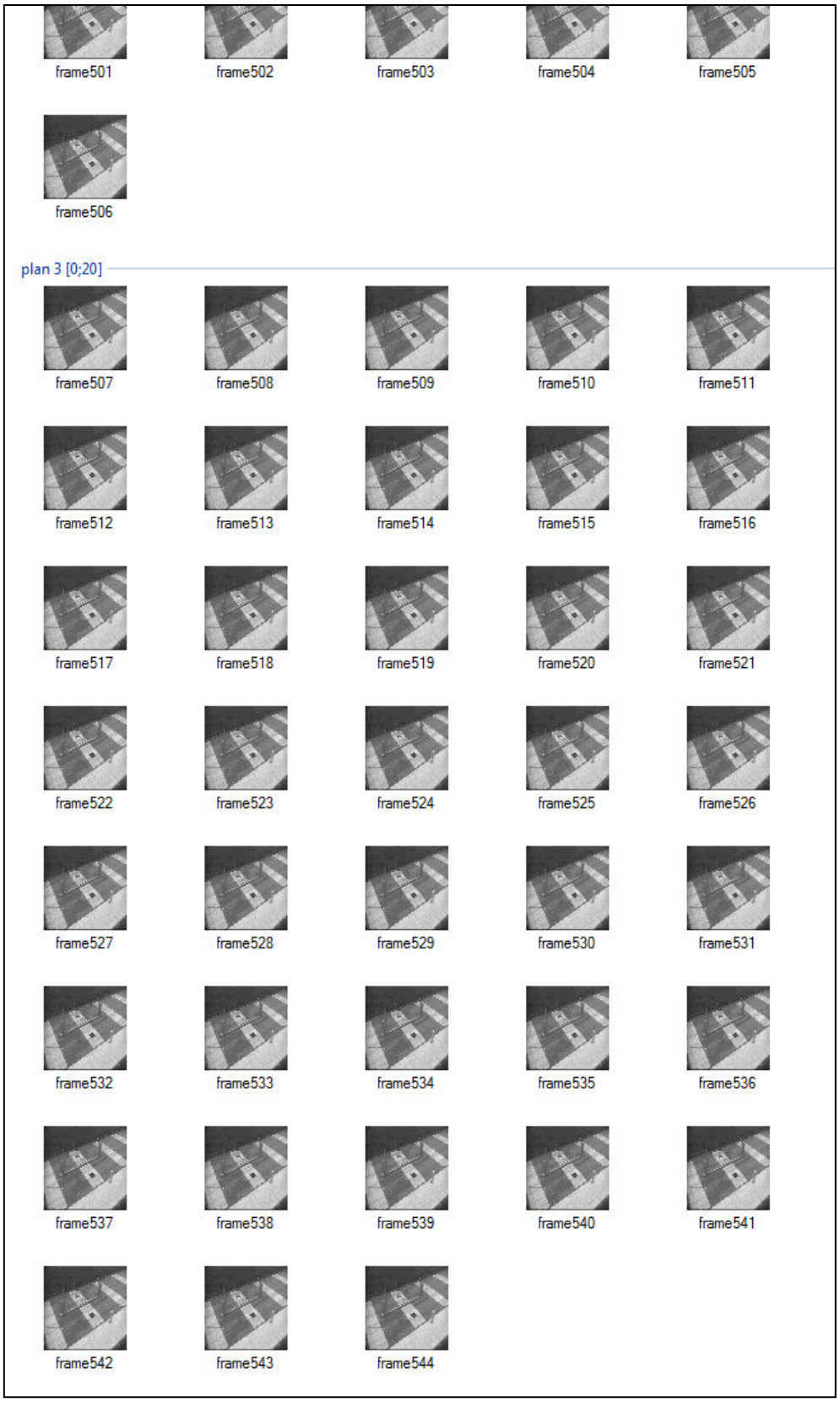


Fig.34. les frames et les plans (501-544)

IV.4.1.b. L'étape d'extraction des keyframes

Dans cette partie nous allons découvrir les keyframes donnés par notre approche. La figure 35 nous montre les keyframes sélectionnés qui sont en fait le premier frame de chaque plan.

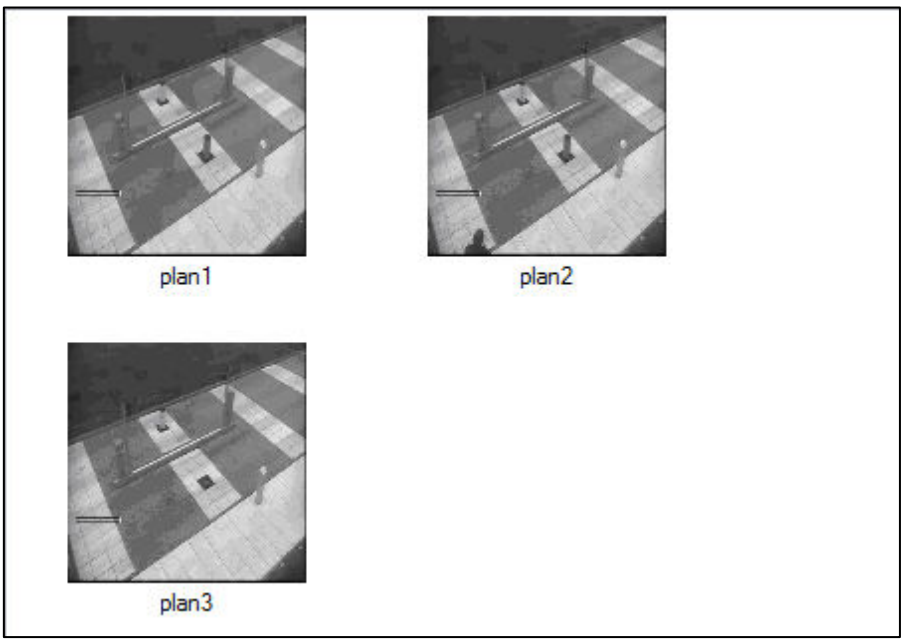


Fig.35.les keyframes

La figure ci-dessous illustre les keyframes après cliquer sur le bouton « régénérer les keyframes».

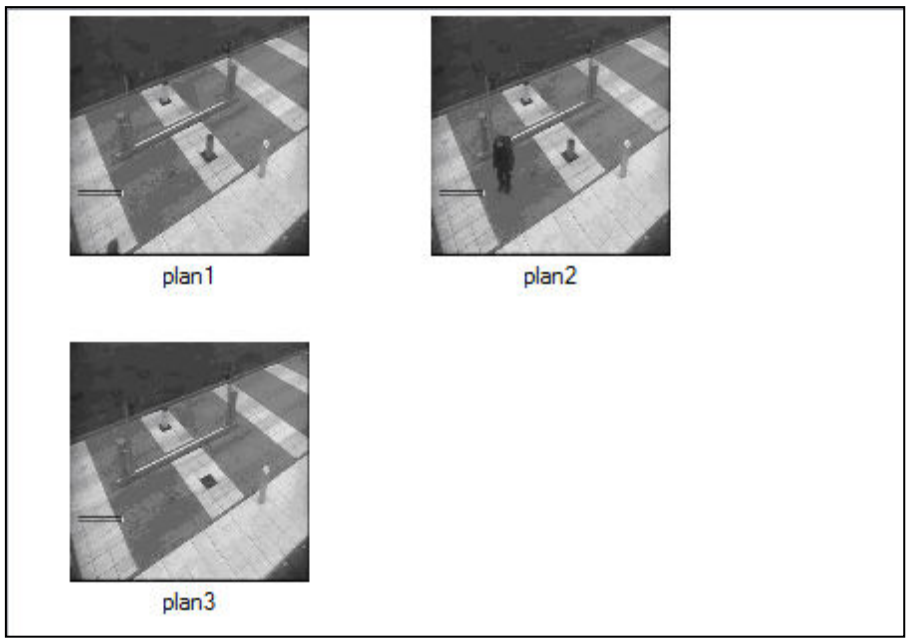


Fig.36.les keyframes réélus

Dans la figure 36 on voit que le deuxième keyframe qui représente le deuxième plan de notre vidéo avait largement changé, on voit que l'homme est clairement apparu par rapport au frame précédent (avant la réélection).

Après avoir notre résultat nous avons décidé d'enregistrer les SIFT des keyframes sous forme TXT, et voila les résultats (figure 37)

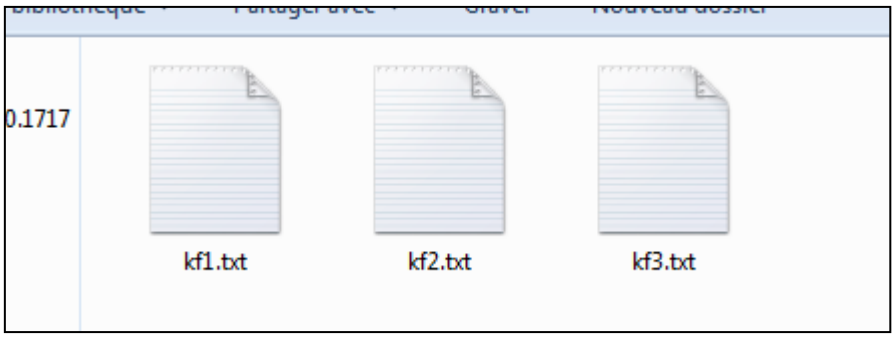


Fig.37.les résultats de l'enregistrement des SIFT des keyframes

IV.4.2. Le deuxième exemple

Notre exemple est le flux vidéo de la webcaméra intégrée de l'ordinateur utilisé pour le développement de l'application.

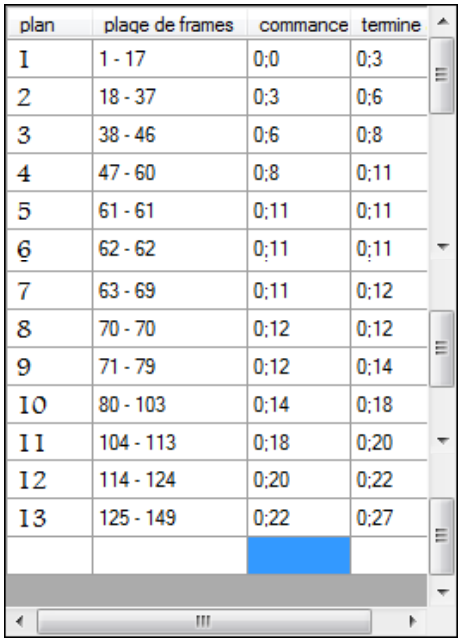
IV.4.2.a. L'étape d'indexation

Comme montre la figure 38 Les résultats de cette étape sont 150 frames qui sont distribués en 13 plans.

Fig.38.le résultat de la deuxième expérimentation

On remarque que 13 plans c'est un peut beaucoup pour une vidéo qui ne contient que 150 frames, cela s'est produit car nous avons volontairement créé un changement de plans pour avoir plus de test dans une petite période de temps.

La figure ci-dessous donne des informations sur les 13 plans qui sont les résultats de notre approche de détection de coupure.



plan	plage de frames	commance	termine
1	1 - 17	0;0	0;3
2	18 - 37	0;3	0;6
3	38 - 46	0;6	0;8
4	47 - 60	0;8	0;11
5	61 - 61	0;11	0;11
6	62 - 62	0;11	0;11
7	63 - 69	0;11	0;12
8	70 - 70	0;12	0;12
9	71 - 79	0;12	0;14
10	80 - 103	0;14	0;18
11	104 - 113	0;18	0;20
12	114 - 124	0;20	0;22
13	125 - 149	0;22	0;27

Fig.39. informations sur les plans de la deuxième expérimentation

Les résultats dans la figure 39 sont les suivants :

- **Le premier plan** commence du 1er frame jusqu'au 17e frame, et de la seconde 0 à la seconde 3.
- **Le deuxième plan** commence du 18e frame jusqu'au 37e frame, et de la seconde 3 à la seconde 6.
- **Le troisième plan** commence du 38e frame jusqu'au 46e frame, et de la seconde 6 à la seconde 8.
- **Le quatrième plan** commence du 47e frame jusqu'au 61 frame, et de la seconde 8 à la seconde 11.
- **Le cinquième plan** commence du 61e frame jusqu'au 61e frame, et de la seconde 11 à la seconde 11 .
- **Le sixième plan** commence du 62e frame jusqu'au 62e frame, et de la seconde 11 à la seconde 11.
- **Le septième plan** commence du 63e frame jusqu'au 69e frame, et de la seconde 11 à la seconde 12.

- **Le huitième plan** commence du 70e frame jusqu'au 70 frame, et de la seconde 12 à la seconde 12.

- **Le neuvième plan** commence du 71e frame jusqu'au 79e frame, et de la seconde 12 à la seconde 14.

- **Le dixième plan** commence du 80e frame jusqu'au 103e frame, et de la seconde 14 à la seconde 18.

- **Le onzième plan** commence du 104e frame jusqu'au 113e frame, et de la seconde 18 à la seconde 20.

- **Le douzième plan** commence du 114e frame jusqu'au 124e frame, et de la seconde 20 à la seconde 22.

- **Le treizième plan** commence du 125e frame jusqu'au 149e frame, et de la seconde 22 à la seconde 27.

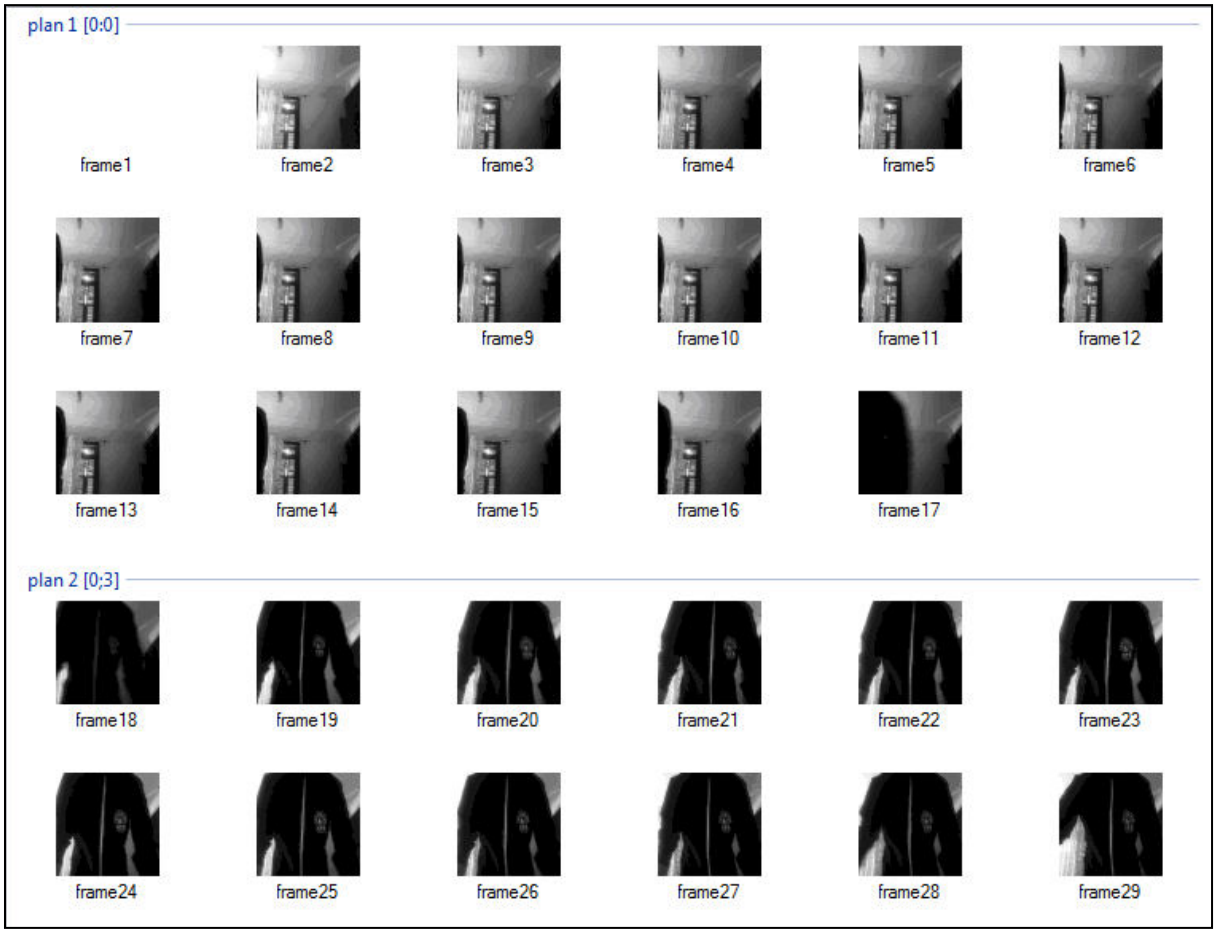


Fig.40. les frames et les plans (1-29)

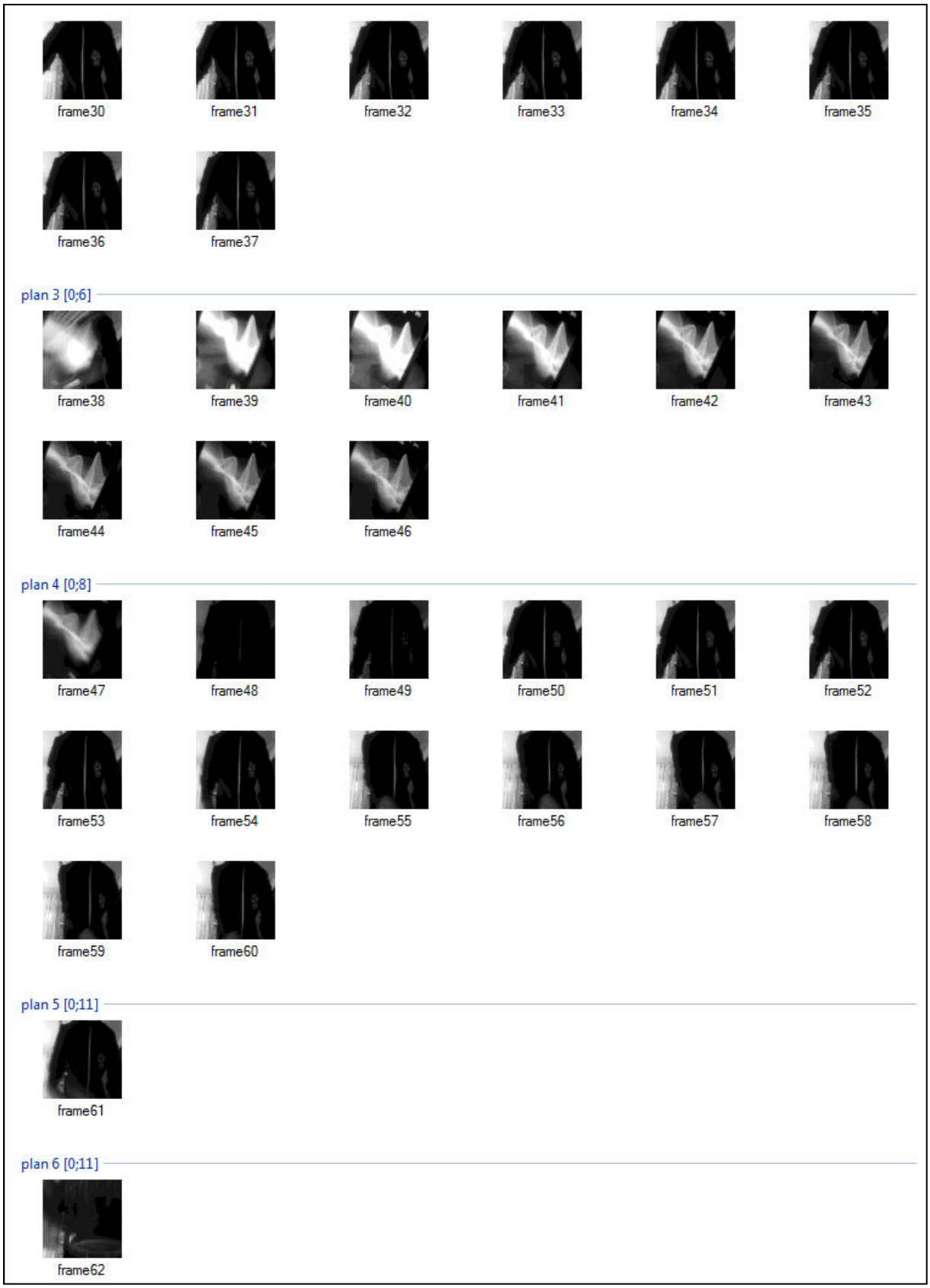


Fig.40. les frames et les plans (30-62)

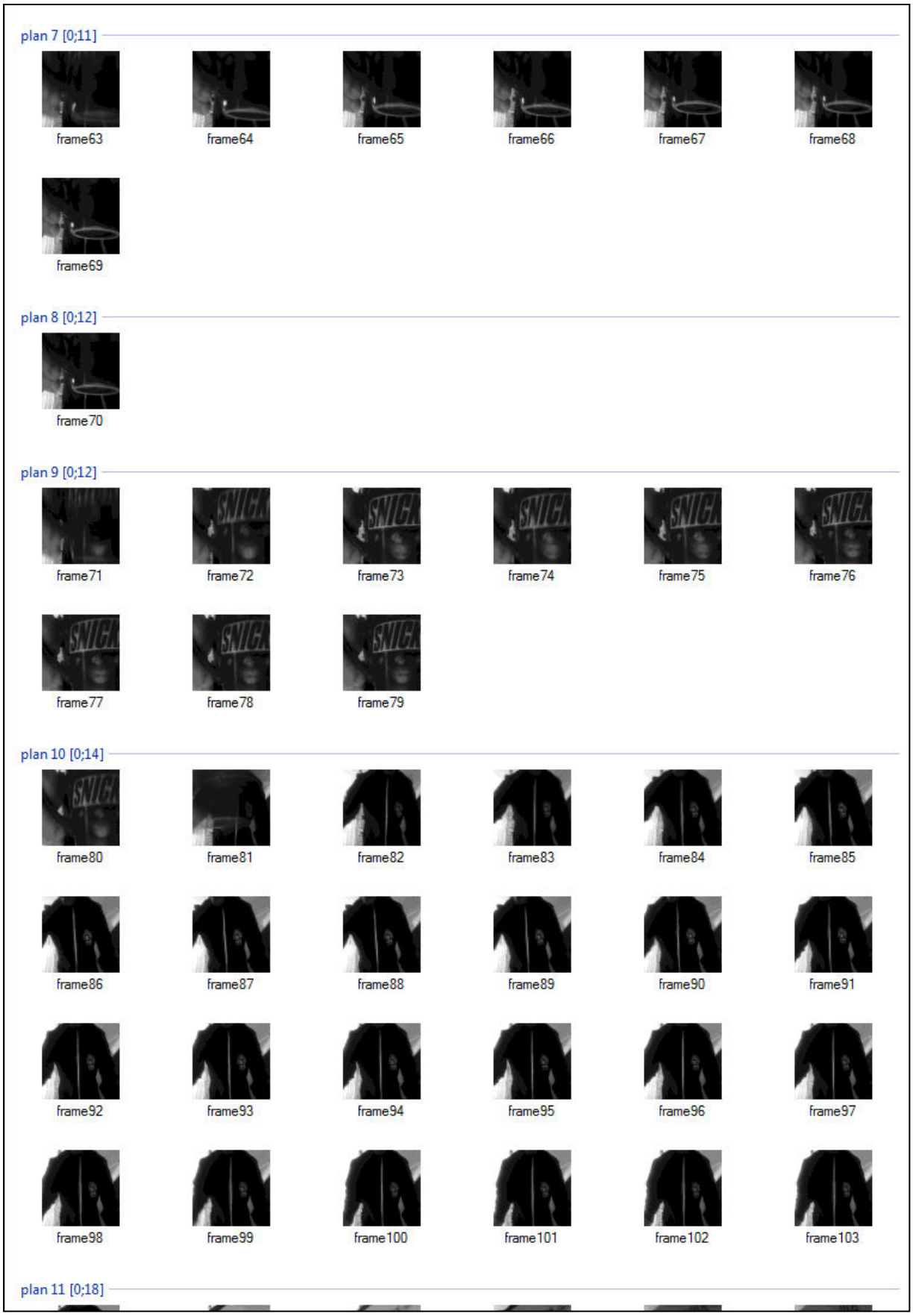


Fig.40. les frames et les plans (63-103)

Fig.40. les frames et les plans (104-150)

IV.4.2.b. L'étape d'extraction des keyframes

Comme nous avons vu dans l'exemple précédent, dans cette étape nous allons exposer les keyframes générés par notre approches.

Fig.41.les keyframes

Tous les keyframes sur la figure précédente « fig 41 » ce sont les premiers frames de chaque plan, nous avons dit que tous les frame d'un plan sont presque similaire , cela dépend du seuil choisi mais pour définir des nouveaux keyframes qui portent des informations importantes on obtient la figure ci-dessous .

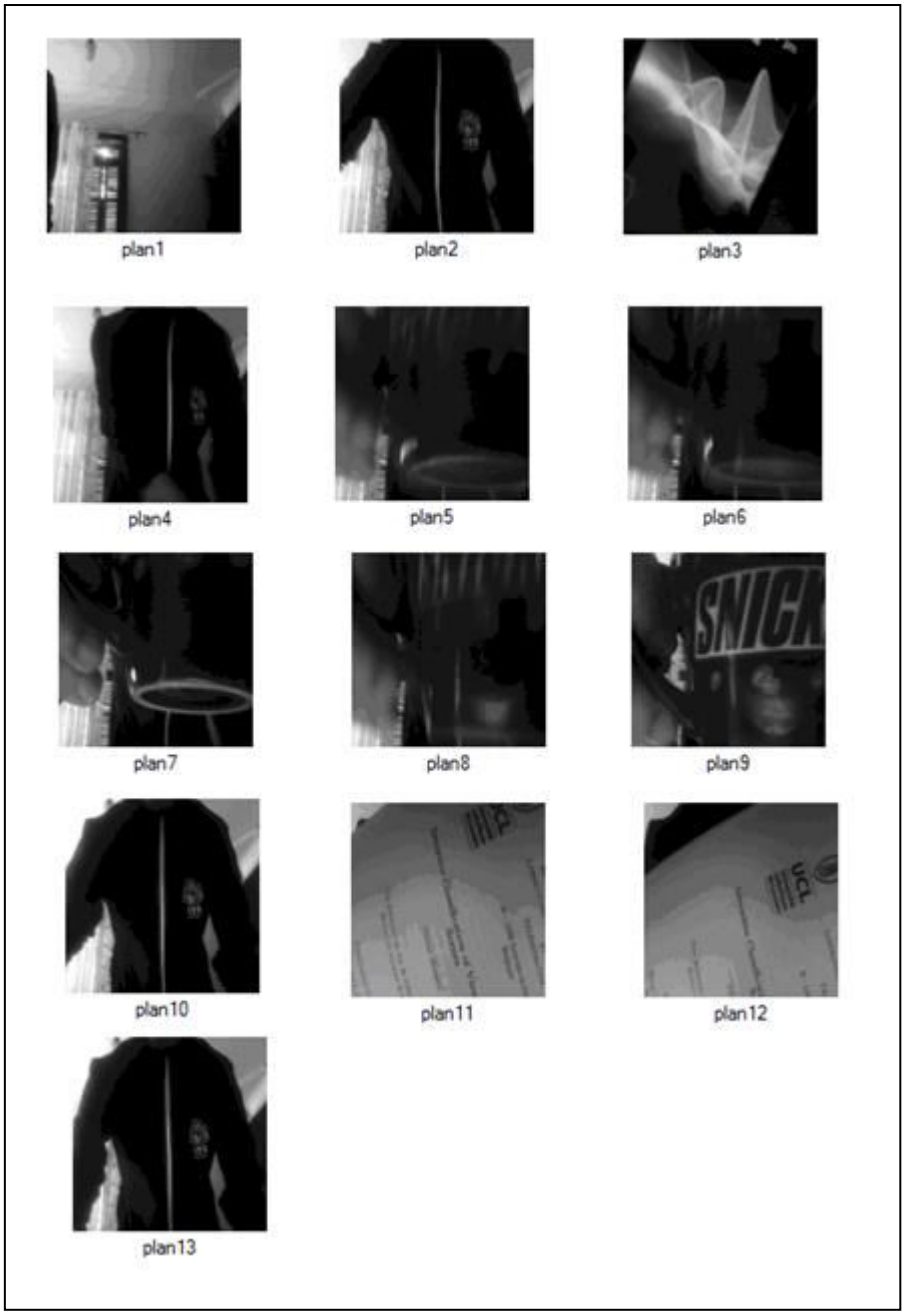


Fig.42.les keyframes réélus

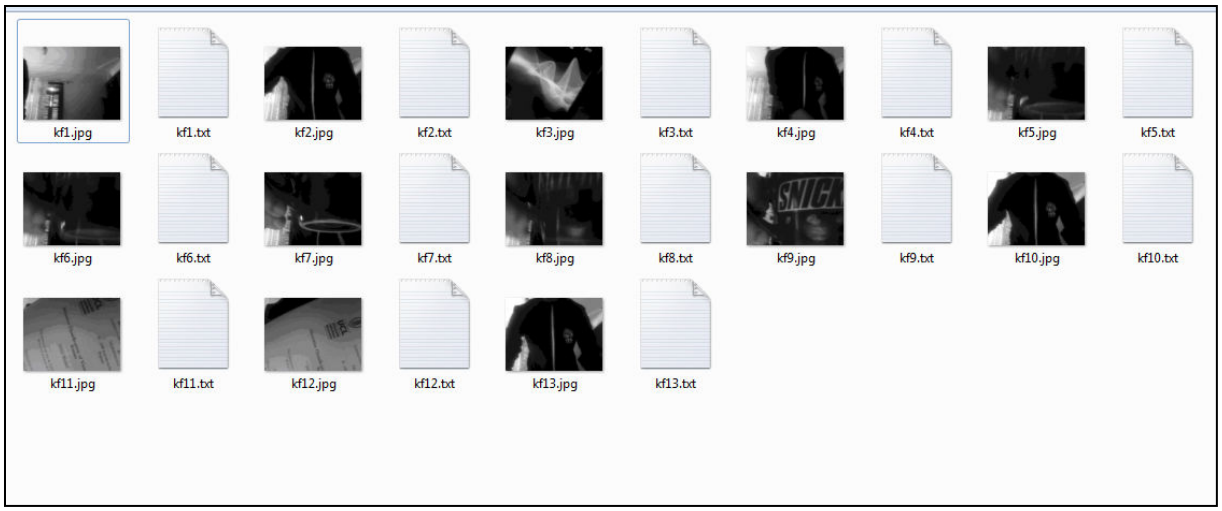


Fig.43.l'enregistrement final des keyframes et leurs SIFTs

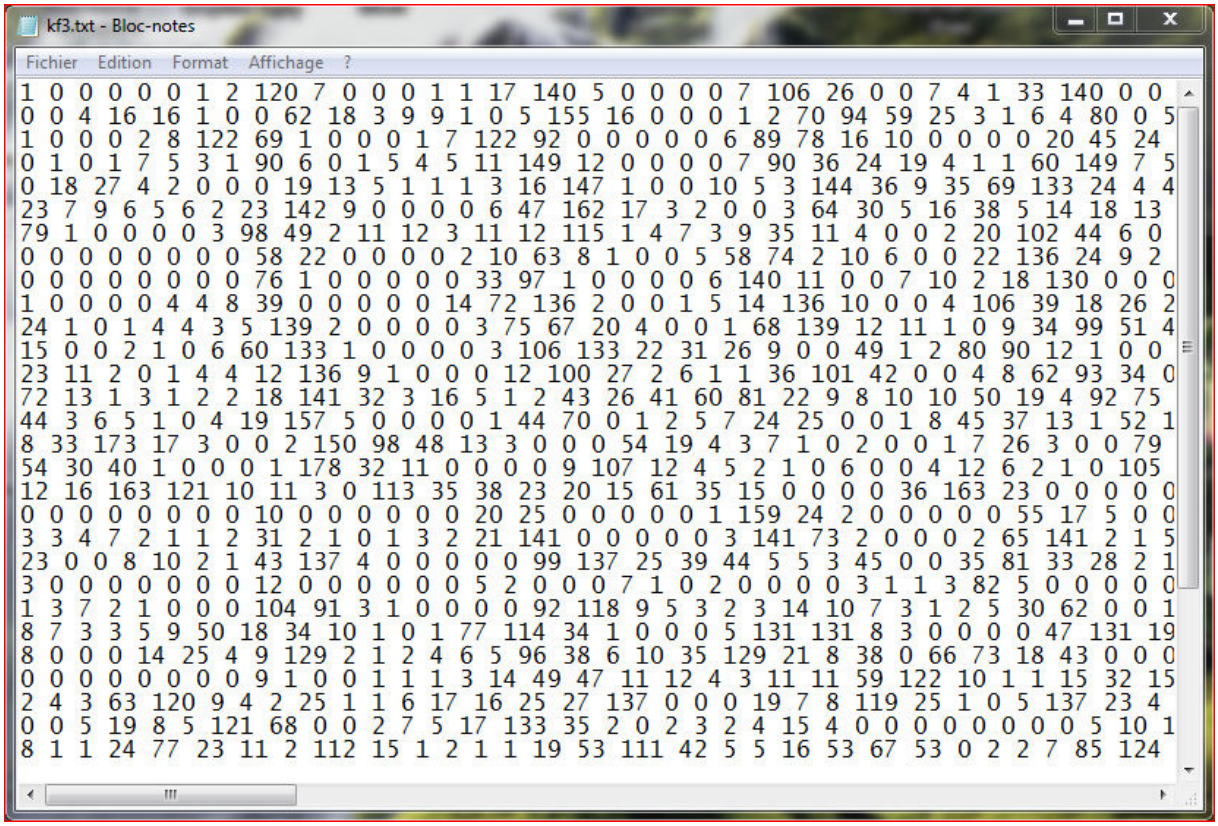


Fig.43.les données SIFT du fichier TXT

IV.5. Performance

Nous avons pris comme test deux exemples, le premier c'est de la vidéo enregistrée et le deuxième est de la webcaméra de notre ordinateur.

On voit que notre système à rendue des bons résultats pour un seuil de valeur 0,25 dans une de 2 minutes et 20 secondes. La distance choisie était la distance euclidienne carrée.

Pour le deuxième exemple la durée du traitement est la même durée de l'exécution du programme, le temps du traitement était 27 secondes dans laquelle nous avons pu capturer 150 frames, ce qui veut dire 6 frames par seconde. Mais par contre les résultats montrés dans la figure 40 étaient acceptables, le seuil était de 0,8 avec la distance de Manhattan.

IV.6. Conclusion

Ce chapitre est consacré à la description et aux performances de notre approche, nous avons fait quelques expérimentations avec des différentes valeurs du seuil, différentes vidéos, différentes captures de caméra, différents choix de distance , les résultat étaient plus au moins satisfaisants.

Conclusion générale

La détection d'une coupure dans une vidéosurveillance est un domaine de recherche actif et important qui a l'objectif de mieux indexer la vidéosurveillance et faciliter la recherche.

Le but de notre projet est de donner un panorama sur la détection d'une coupure dans une vidéo surveillance afin d'avoir la capacité de l'indexer. Tout d'abord, nous avons défini le concept d'une donnée vidéo, par la suite nous avons fais un bref bilan sur les systèmes d'indexation et de recherche par le contenu.

Tant que nous avons évoqué l'indexation et la recherche par le contenu, nous avons décrit que ce que c'est l'extraction des caractéristique visuelle en donnant ses différents types et les méthodes les plus utilisées.

Pour mieux exposer notre projet, nous avons détaillé le principe du déroulement de notre approche et ses résultats, on constate qu'avec nos ressources simples nous avons obtenu des résultats bons et même la capture en temps réel était commodément convenable.

Bibliographie

[1] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, '' Speeded Up Robust Features'', Katholieke Universiteit Leuven.

[2] Pascal Kelm, Sebastian Schmiedeke, and Thomas Sikora ,FEATURE-BASED VIDEO KEY FRAME EXTRACTION FOR LOW QUALITY VIDEO SEQUENCES,Communication Systems Group, TU Berlin, EN-1, Einsteinufer 17, 10587 Berlin, Germany.

[3] Masahito Kumano and Yasuo Ariki , Automatic Useful Shot Extraction for a Video Editing Support System ,Faculty of Science and Technology ,Ryukoku University.

[4] Herbert Bay a , Andreas Ess a , Tinne Tuytelaars b , and Luc Van Gool a;b Speeded-Up Robust Features (SURF) , K. U. Leuven, ESAT-PSI Kasteelpark Arenberg 10 B-3001 Leuven Belgium.

[5] Alan C. Bovik ,the essential guide to image processing, diacriTech, India ; (2009).

[6] Khouloud Meskaldji, Samia Boucherkha et Salim Chikhi , Color Quantization and its Impact on Color Histogram Based Image Retrieval, Networked Digital Technologies, 2009. NDT '09. First International Conference on IEEE.

[7] Mark Ewald, Content-Based Image Indexing and Retrieval in anImage Database for Technical Domains;Institute of Computer Vision and Applied Computer Sciences Kohlenstr. 2, 04109 Leipzig, Germany ;2009.

[8] R.BALU M.Sc., M.Phil, Devi, T ;Design and development of automatic appendicitis detection system using sonographic image mining; Department of Computer Science, Bharathiar University ; 2012.

[9] A.Ramesh Kumar, D.Saravanan , Content Based Image Retrieval Using Color Histogram,*Sathyabama University, Chennai ,Tamil Nadu, India.*

[10] David G. Lowe; Distinctive Image Features from Scale-Invariant Keypoints; Computer Science Department, University of British Columbia, Vancouver, B.C., Canada; (January 5, 2004).

[11] ZHU CHAOYANG, Video Object Tracking using SIFT and Mean Shift ;CHALMERS UNIVERSITY OF TECHNOLOGY Göteborg, Sweden ; 2011.

[12]R. Brunelli and O. Mich and C. M. Modena; A Survey on the automatic Indexing of video data ;journal of visual communication and image representation;(1997)

[13] Dian W. Tjondronegoro ; Content-based Video Indexing for Sports Applications using Integrated Multi-Modal Approach, Deakin University May ;2005.

[14] Onur Küçüktunç, Ugur Güdükbay , Özgür Ulusoy;Fuzzy color histogram-based video segmentation;Bilkent University, Department of Computer Engineering, Bilkent, 06800 Ankara, Turkey;press of Computer Vision and Image Understanding (2009).

[15] Noor A. Ibraheem, Mokhtar M. Hasan, Rafiqul Z. Khan, Pramod K. Mishra;Understanding Color Models;ARPN Journal of Science and Technology;(2012)

[16] Dong ping Tian;A Review on Image Feature Extraction and Representation Techniques; International Journal of Multimedia and Ubiquitous Engineering;Vol. 8, No. 4, July, 2013.

[17]M. Guironnet, D. Pellerin, P. Ladret ; Combining color and activity fuzzy descriptors for vidéo summaries ; 2002.

[18] B.S. Manjunath*, P. Wu, S. Newsam, H.D. Shin1;A texture descriptor for browsing and similarity retrieval;Department of Electrical and Computer Engineering, University of California.

[19]Jhon F, David reinsel, Christopher chute,Wolfgang Schlicht, John McArthur, Anna Toncheva et Alex Manferdiz: tthe Expanding Digital Universe, a Forecast of Information Through 2010".

[20] Navneet Dalal and Bill Triggs , Histograms of Oriented Gradients for Human Detection, INRIA Rh.one-Alps, 655 avenue de l'Europe, Montbonnot 38334, France.

[21] Longin Jan Latecki and Rolf Lakamper and Ulrich Eckhardt;Shape Descriptors for Non-rigid Shapes with a Single Closed Contour;IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 424-429, 2000.

[22] Per-Erik Forss´en and David G. Lowe; Shape Descriptors for Maximally Stable Extremal Regions;Department of Computer Science University of British Columbia.