AGENCE UNIVERSITAIRE DE LA FRANCOPHONIE (AUF)

INSTITUT FRANCOPHONE INTERNATIONAL (IFI)

¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤¤

MATIÈRE

RECONNAISSANCE DE FORME

RAPPORT FINAL

PROJET 2: ETUDES ET EXPÉRIMENTATION DE LA CLASSIFICATION DES SCÈNES NATURELLES

Réalisé par :
KOBINA Piriziwè, Promo 20
MEDOU Daniel Magloire, Promo 21
NGASSA TCHOUDJEUH Tatiana, Promo 20

Enseignant:
Dr HO Tuong Vinh

Année Académique 2016 - 2017

SOMMAIRE

INTRODUCTION

La compréhension de la scène est une caractéristique de la vision naturelle humaine et constitue un objectif difficile pour la vision mécanique car une scène contient des informations prédictives sur de multiples échelles de traitement. Les modèles informatiques de la compréhension des scènes ont tenté d'identifier les signatures des scènes et de les utiliser pour la classification de l'image. En effet, Les humains sont extrêmement compétents pour percevoir des scènes naturelles et comprendre leur contenu. Cependant, nous savons très peu sur la façon dont, ou même à quel niveau du cerveau, nous traitons les scènes naturelles. Comment est-ce que le cerveau détermine s'il regarde la plage ou un horizon de la ville par exemple?
Comment les humains reconnaissent-ils rapidement une scène? Comment les modèles neuronaux peuvent-ils saisir cette compétence biologique pour obtenir une classification de scène à la fine pointe de la technologie? Nous présenterons, d'abord, les quelques méthodes existantes permettant de faire la classification des scènes naturelles, ensuite, nous proposons une solution que nous implémenterons afin de faire la classification des scènes naturelles et enfin, nous présenterons les différents résultats obtenus à partir de notre solution proposée et implémentée.

I- ETAT DE L'ART SUR LA CLASSIFICATION DES SCÈNES NATURELLES

Dans la phase de l'état de l'art, nous avons étudié et présenté deux techniques ou approches de la classification de scènes naturelles à savoir ARTSCENE et l'approche Bag-of words.

1- ARTSCENE: Un système neuronal pour la classification des scènes naturelles

La représentation des scènes naturelles est susceptible de résider simultanément à une échelle spatiale fine, par exemple les neurones sensibles aux plages et les villes peuvent être intercalés les uns avec les autres, et être distribués dans le cortex. Ainsi, le système neuronal ARTSCENE [1] classe les photographies de scène naturelles en utilisant des échelles spatiales multiples pour accumuler efficacement des preuves des principaux points et de la texture.

Pour étudier comment les humains analysent une scène dans des éléments locaux, les auteurs sur ARTSCENE ont utilisé des annotations humaines sur un ensemble de données d'images que certains chercheurs avaient aussi utilisé, qui sont disponibles à partir du site Web *LabelMe*. Bien que ce plan d'annotation incorpore les coordonnées des polygones et les noms des étiquettes des régions locales, ce n'est pas un ensemble de données sans erreur pour la classification des textures, car son problème majeur est la mauvaise la segmentation des images. Un problème connexe est que les noms d'étiquettes sont ambigus s'ils sont pris localement sans contexte. C'est l'exemple où une étiquette "**Eau**" peut inclure un ciel et des montagnes en raison de la réflexion (*fig. 1*), et une étiquette "**Pierre**" peut être confondue avec des nuages en raison de l'occlusion (*fig. 2*). Ci-dessous les images illustratives de ces différents problèmes que nous venons d'evoquer:

***Fig. 1*: Illustration de l'étiquetage avec réflexion**

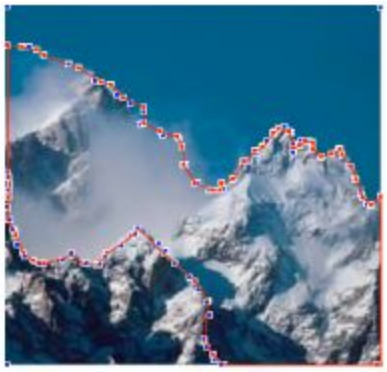


***Fig. 2*: Illustration de l'étiquetage avec occlusion**

Le système ARTSCENE possède deux modèles à savoir un modèle d'apprentissage et un modèle de test que nous illustrons ci-dessous:

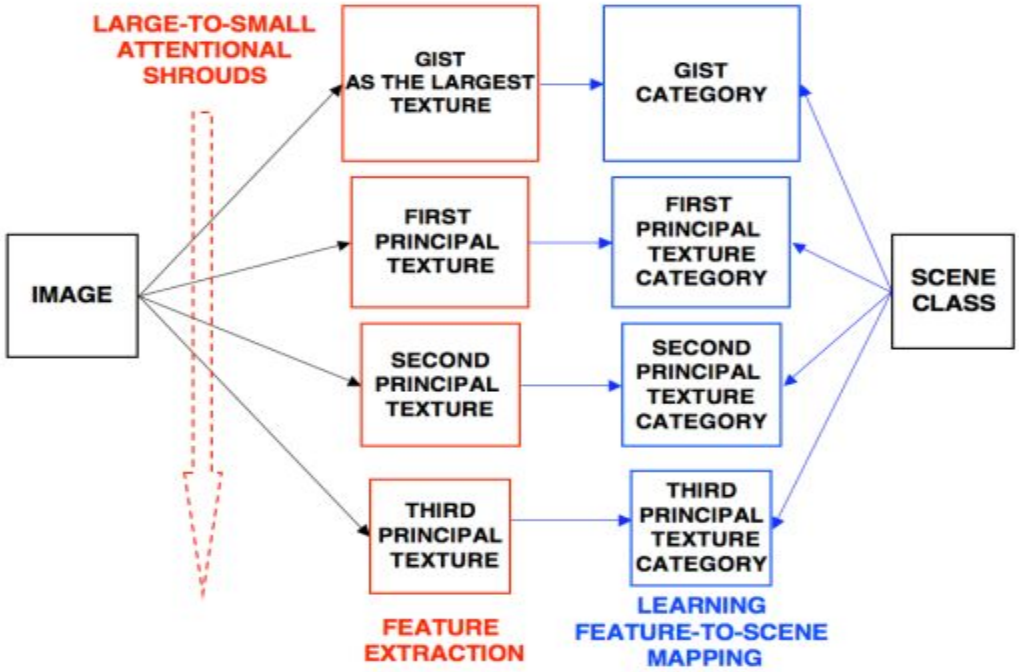


***Fig. 3* : Modèle d'apprentissage**

Le modèle d'apprentissage est constitué d'une architecture neuronale à deux couches dont la première couches consiste à l'extraction des caractéristiques et la seconde à l'apprentissage des caractéristiques extraites pour la classification.

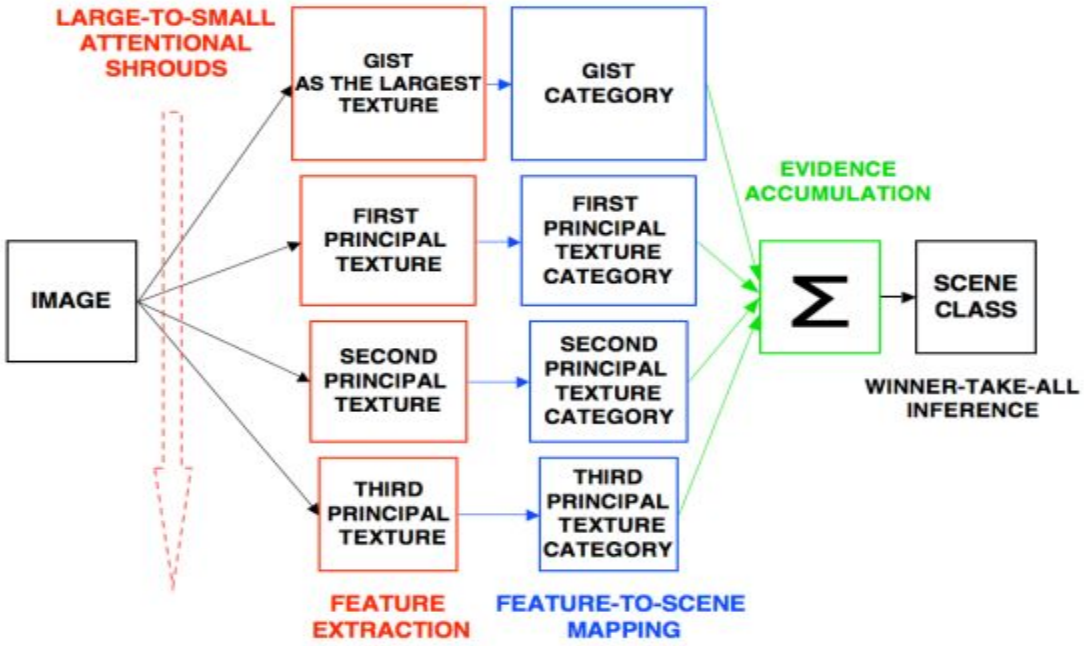


***Fig. 4*: Modele de test**

Tout comme le modèle d'apprentissage, celui de test possède est aussi constitué d'une architecture à deux couches. La particularité du modèle de test est qu'il possède un point d'activation de la sortie. Ce point d'activation de la sortie permet d'accumuler les preuves pour les principaux points et sur la texture des images.

2- Algorithme amélioré des sacs de mots (Bag of Word) pour la reconnaissance de la scène

La technique de sac de mots (Bag of Word en anglais) est aussi appliquée pour la reconnaissance de scène. Dans [2], la méthode du BoW, version améliorée nous est présentée comme une technique très efficace pour la reconnaissance de scène effectuant une meilleure classification. Dans le cadre de cette technique de BoW, l'algorithme **K-means** est utilisé pour regrouper tous les vecteurs caractéristiques de l'image en un certain nombre de grappes (cluster), dans lequel chaque mot de code est représenté par le centroïde du cluster et en plus d'une matrice d'occurrence. **Modèle Mixte Gaussien (GMM)** est ensuite utilisé par la technique du BoW pour modéliser la distribution de chaque cluster que le K-means a généré. Alors, le GMM est considéré comme "mot de code" du codebook qui est représenté par le centroïde du cluster. Enfin, un histogramme BoW est établie pour représenter chaque image grâce à l'affectation souple des fonctions d'image à chaque GMM. Pour former ces histogrammes BoW, la technique utilise classificateur **SVM (machine vectorielle de support)** pour la correspondance des images**.** La figure ci-dessous est une illustration d'un système typique de reconnaissance de scène.

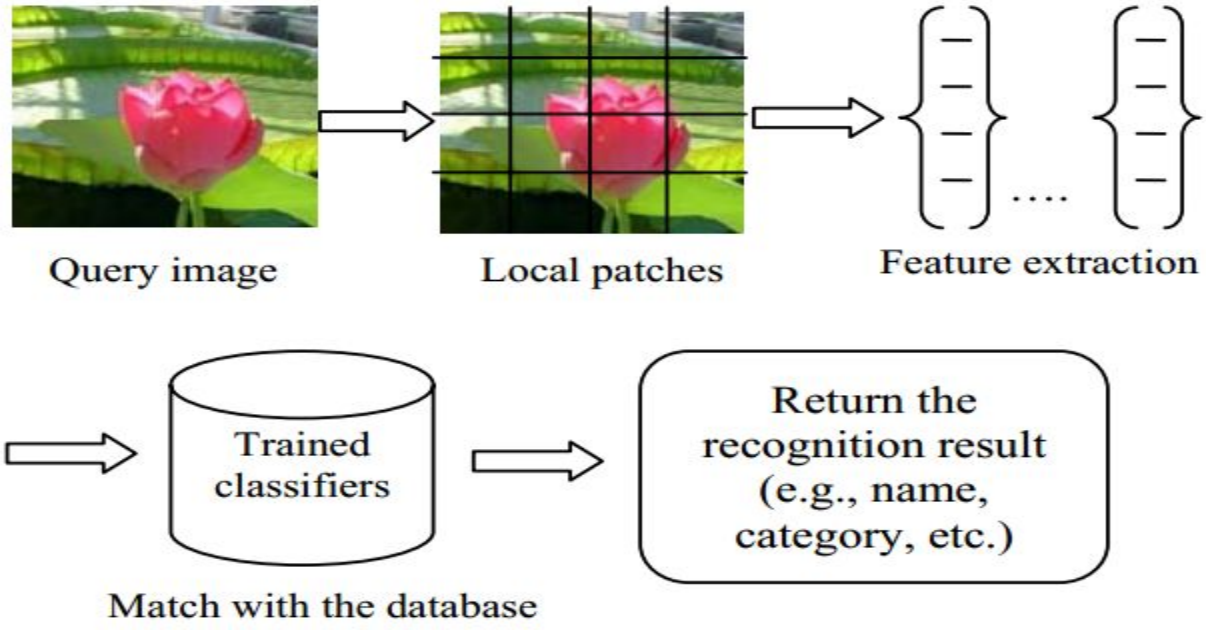


Fig.5: Illustration d'un système de reconnaissance de scène

La technique du BoW, basée sur l'algorithme K-means et le GMM avait été expérimentée quinze (15) jeux de données de scène pour l'évaluation de la méthode. La machine de vecteur de support (SVM) avec un noyau d'intersection de l'histogramme en tant que classificateur, est utilisée. 300 clusters sont utilisés comme taille de code. Les résultats finaux sont présentés dans le tableau ci-dessous:

Méthode d'apprentissage codebook	Méthode d'établissement de l'histogramme	Précision de reconnaissance (%)
Méthode en 0	Méthode en 0	75
Méthode en 0	Méthode en 0	79
Méthode basée sur GMM	Affectation difficile dans 0	77
Méthode en 0	Histogramme doux proposé	76
Méthode basée sur GMM	**Histogramme doux proposé**	**81**

II- SOLUTION PROPOSEE

Pour la réalisation de notre système de classification des scènes naturelles, nous nous sommes proposés d'implémenter l'approche Bag-of-Words, en abrégé BoW. L'approche Bag-of-Words, est une description de document. Pour les images, le dictionnaire est généralement composé de caractéristiques locales. On parle alors de sac de mots visuels. Le sac de mots visuels est aussi une représentation des images pouvant être utilisée dans le cadre de la classification supervisée. L'intérêt premier est qu'une image représentée originellement par un nombre variable de caractéristiques locales est ramenée dans un espace vectoriel de dimension fixe, et peut ainsi "alimenter" un algorithme d'apprentissage. Nous verrons plus loin l'architecture de cette approche.

Les descripteurs locaux d'une image sont des caractéristiques locales calculées autour des points d'intérêts de l'image. Les caractéristiques locales se distinguent par le fait qu'elles sont distinctes et robustes aux occlusions, car il y en a beaucoup dans une image ou une région de l'image et qu'elles ne nécessitent pas de segmentation de l'image. Ainsi, un descripteur local est calculé en chaque pixel de l'image. Les points d'intérêts sont des points dans l'image qui définissent ce qui est intéressant ou pertinent dans l'image. La raison pour laquelle les points d'intérêts sont spéciaux est que, peu importe la transformation de l'image, c'est-à-dire si l'image subi une rotation, se rétrécit ou se dilate ou encore subi une distorsion, l'on devrait trouver les mêmes points d'intérêts dans cette image modifiée lorsqu'on la compare avec l'image originale.

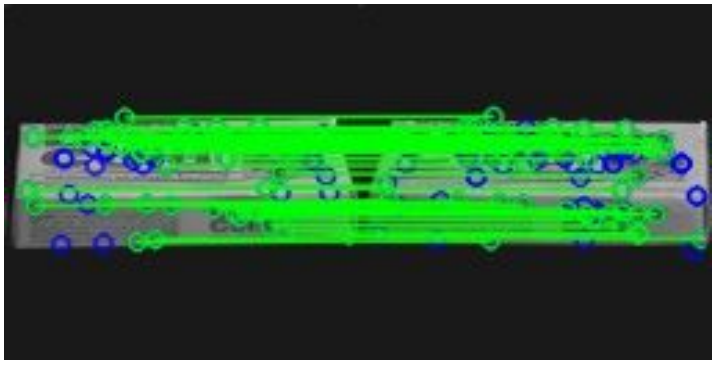


***Fig.6*: Illustration des points d'intérêts et des descripteurs**

1- Contexte du problème à résoudre

De nombreux chercheurs ont pour but d'étudier et de comprendre comment l'être humain identifie son environnement visuel et d'en tirer une méthode exploitable par une machine. Cette prise en compte de la perception visuelle chez l'homme est une chose nouvelle dans le domaine des technologies de l'information et fait appel à différentes disciplines que sont la psychologie, les neurosciences et les sciences de l'ingénieur. Dans notre cas, nous nous intéressons aux scènes naturelles. Ainsi le développement croissant de nombreuses bases d'images distribuées à travers Internet, les musées, les agences de presse ou de publicité et la demande croissante de moteurs de recherche efficace requiert des outils automatiques de catégorisation, de labellisation et d'indexation d'images. Automatiser ces tâches est actuellement un défi technologique qui mérite d'être relevé.

2- Analyse des données

Dans le cadre de notre projet, nous utiliserons une base d'images constituée de 3859 images de scènes ou catégories différentes. En tout nous disposons de 13 classes ou catégories d'images. Chaque catégorie constitue un dossier et toutes les catégories sont regroupées dans un seul dossier nommé ***SceneClass13***. Certaines images sont en niveau de gris (noir et blanc) et d'autres en couleur.

3- Architecture de la solution proposée

Afin de mieux définir le fonctionnement de notre solution proposée, nous l'avons illustrée par une architecture ci-dessous que nous décrirons dans la suite de ce document.

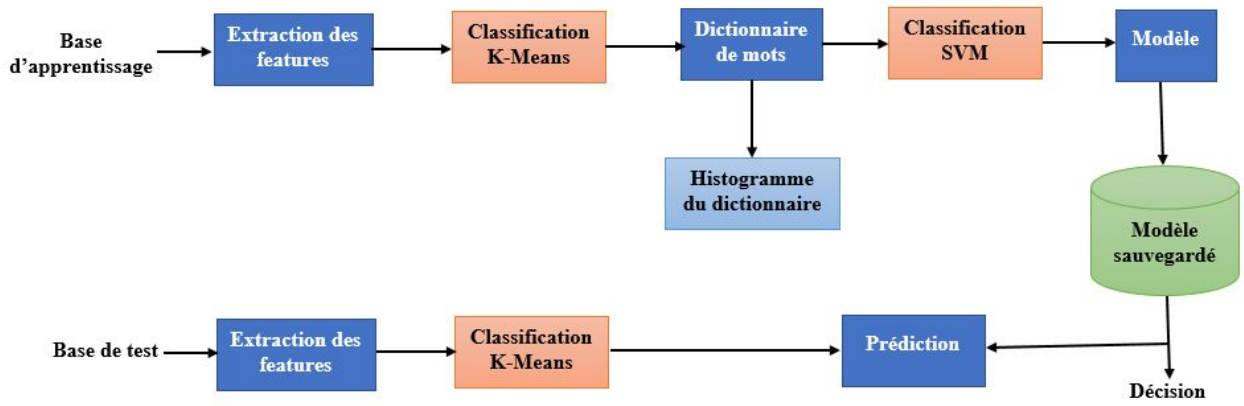


Fig. 7*:*Architecture de notre système

Notre solution fonctionne en deux phases: la phase d'apprentissage et celle de test.

- **Phase d'apprentissage**: notre solution procède à l'acquisition des images d'apprentissage puis à l'extraction des caractéristiques pertinentes (les points d'intérêts et descripteurs). Après cette phase d'extraction des caractéristiques, nous faisons la classification des descripteurs avec l'algorithme de K-Means pour obtenir le dictionnaire de mots. Nous pouvons établir l'histogramme de ce dictionnaire pour voir la fréquence d'apparition des différents mots. La taille du dictionnaire est égale au nombre de classes lors de la classification. Nous faisons ensuite une classification et un apprentissage avec l'algorithme SVM pour construire le modèle d'apprentissage que nous stockons pour une utilisation ultérieure. Nous sauvegardons aussi les différentes caractéristiques extraites pour gagner du temps lors de la prochaine exécution du programme.
- **Phase de test**: pendant cette phase, nous procédons à l'acquisition des images servant de test puis à l'extraction des caractéristiques que nous sauvegardons tout comme lors de la phase d'apprentissage. Après l'extraction des caractéristiques, nous faisons une classification avec l'algorithme de K-Means. Nous pouvons aussi construire un histogramme de chaque image. Cet histogramme est la fréquence d'apparition des mots de chaque image par rapport au dictionnaire. Ensuite nous faisons une prédiction de la classe de chaque image par rapport au modèle construit et sauvegardé lors de la phase d'apprentissage.

4- Outils et évaluation de la solution

Dans le cadre de l'implémentation de notre solution, nous utiliserons ***Python*** comme langage de programmation et la bibliothèque multiplateforme ***OpenCV***. Pourquoi ces choix? La raison est simple car, Dans [L1], ***OpenCV*** se concentre principalement sur le traitement d'image, la capture vidéo et l'analyse, y compris des fonctionnalités telle que la détection d'objets. Et dans notre projet nous allons manipuler les images d'où la nécessité de cette bibliothèque. En ce qui concerne ***Python***, dans [L2], c'est un langage de programmation généralisé, interactif, orienté objet et de haut niveau et conçu pour être très lisible. Ce langage a plusieurs caractéristiques à savoir: la facilité d'apprentissage, la

facilité de lecture, la facilité de maintenance, l'extensibilité, la portabilité, l'évolutivité, etc.

En ce qui concerne l'évaluation de notre système que nous mettrons en place, elle se fera autour des points suivantes: la validation croisée, le calcul de la précision, le calcul du rappel. La précision donne le pourcentage des réponses correctes tandis que le rappel donne le pourcentage des réponses correctes qui sont données.

III- IMPLÉMENTATION DE LA SOLUTION

1- Implémentation

Nous considérons les 100 premières images de chaque catégorie pour constituer la base d'apprentissage soit une base de 1300 images et le reste constituera la base de test soit une base de 2559 images. Tout ceci se fait automatiquement par une de nos fonctions.

Comme nous l'avions défini plus haut, le dictionnaire pour une image est un ensemble de caractéristiques locales.

Les descripteurs SIFT sont des vecteurs à 128 dimensions, de sorte que nous pouvons simplement créer une matrice avec chaque descripteur SIFT dans notre ensemble d'apprentissage comme sa propre ligne et 128 colonnes pour chacune des dimensions des descripteurs SIFT. Nous utilisons cette matrice dans un algorithme de clustering tel que K-Means pour obtenir les descripteurs classifiés en K différents mots c'est-à-dire K groupes ou classes différentes.

Ensuite, nous passons par chaque image individuelle, et assignons tous ses descripteurs SIFT au mot du dictionnaire auquel ils appartiennent. Enfin, nous créons un histogramme pour chaque image en sommant le nombre de caractéristiques pour chaque mot visuel. Ci-dessous l'histogramme d'une image test:

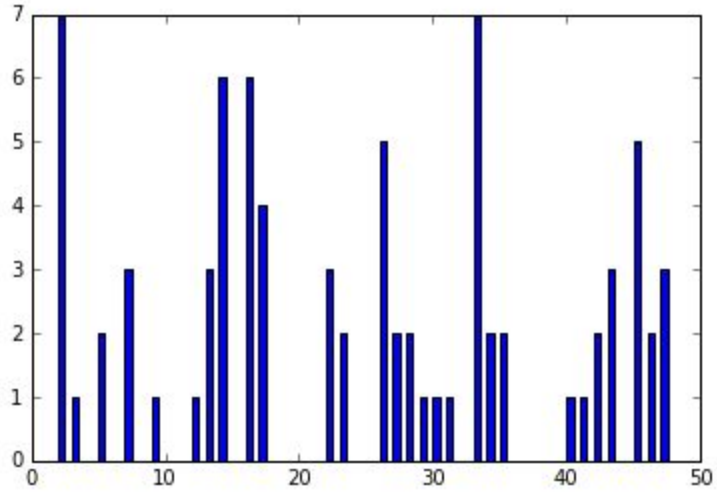
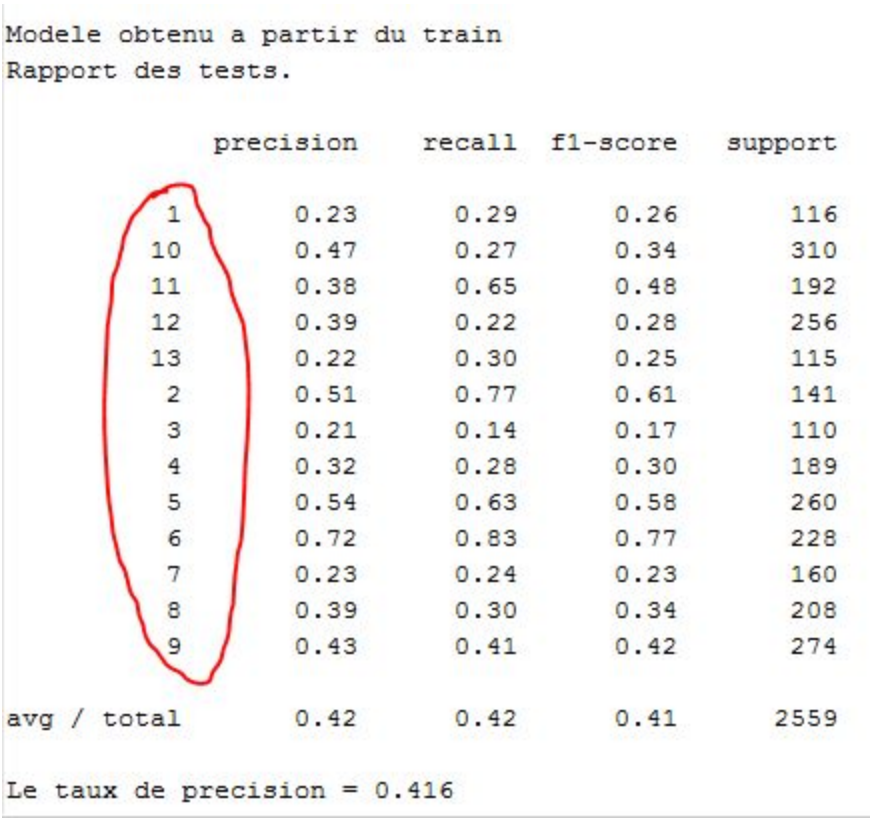


Fig. 8:Fréquence d'apparition des caractéristiques locales d'une image

2- Analyse des résultats

Lors de notre implémentation, nous avons utilisé les descripteurs SIFT afin d'atteindre notre objectif qui était la classification des différentes catégories d'images dont nous disposons. Mais après un certain nombre d'expérimentations, nous obtenons un mauvais taux de classement (la précision) qui est de **42%**. Ci-dessous une illustration de cette expérimentation:



Modele obtenu a partir du train
Rapport des tests.

	precision	recall	f1-score	support
1	0.23	0.29	0.26	116
10	0.47	0.27	0.34	310
11	0.38	0.65	0.48	192
12	0.39	0.22	0.28	256
13	0.22	0.30	0.25	115
2	0.51	0.77	0.61	141
3	0.21	0.14	0.17	110
4	0.32	0.28	0.30	189
5	0.54	0.63	0.58	260
6	0.72	0.83	0.77	228
7	0.23	0.24	0.23	160
8	0.39	0.30	0.34	208
9	0.43	0.41	0.42	274
avg / total	0.42	0.42	0.41	2559

Le taux de precision = 0.416

<u>Fig. 9</u>: Illustration de l'expérimentation donnant un mauvais taux de précision

La partie partie encerclée au rouge représente les différentes classes ou catégories numérotée de **1** à **13**

Nous avons donc conclu, à la suite de l'obtention de ce mauvais résultat, que le descripteur utilisé n'était pas bon pour nos données.

Ensuite, nous avons essayé d'améliorer notre programme en changeant de descripteurs. Nous avons utilisé les points d'intérêts dont la taille est de 4 c'est-à-dire que nous extrayons seulement quatre (04) points d'intérêts pertinents sur chaque image, ceci pour réduire la consommation en mémoire de la machine. Cette amélioration nous a permis d'avoir un taux de précision de **80%**. Ci-dessous les résultats obtenus suite à cette expérimentation:

```
Modele obtenu a partir du train
Rapport des tests.

             precision    recall  f1-score   support

          1       0.76      0.91      0.84       116
         10       0.80      0.85      0.83       310
         11       0.77      0.97      0.87       192
         12       0.82      0.89      0.86       256
         13       0.60      0.77      0.69       115
          2       0.82      0.73      0.78       141
          3       0.76      0.89      0.83       110
          4       0.86      0.99      0.93       189
          5       0.76      0.85      0.81       260
          6       0.95      0.64      0.80       228
          7       0.73      0.85      0.79       160
          8       0.89      0.93      0.91       208
          9       0.72      0.71      0.72       274

avg / total       0.79      0.80      0.80      2559

Le taux de precision = 0.796
```

***Fig. 10*: Illustration de l'amélioration du programme**

Nous avons, ensuite, essayé d'utiliser un descripteur basé sur l'aspect géométrique des images. Ce descripteur est le ***Spatial Pyramid Matching*** (SPM). Avant l'utilisation de SPM, nous avons extrait les caractéristiques avec SIFT, caractéristiques utilisées pour améliorer le résultat. Mais lors de l'exécution, le programme renvoie une erreur de mémoire. Ceci est dû au traitement d'un grand volume de données par rapport à nos machines dont les caractéristiques ou ressources sont insuffisantes.

CONCLUSION

L'analyse de scènes complexe est un des domaines les plus difficiles et les plus étudiés en vision par ordinateur selon une approche complexe ascendante. Du pixel au groupe de pixels, du groupe de pixels aux traits, du trait aux objets, et des objets à la scène, les stratégies sont nombreuses et se basent toutes sur les techniques diverses.

Ainsi, dans le cadre de notre projet, nous avons mis en place un système de classification de scènes naturelles qui nous a permis d'obtenir un taux de classement ou une précision de **80%**. avant tout, nous avons effectué une bibliographie des techniques et méthodes existantes puis proposé une solution que nous avons ensuite implémenté.

REFERENCES

[1]: Stephen Grossberg & Tsung-Ren Huang, "***ARTSCENE: A Neural System for Natural Scene Classification***", Submitted: September 27, 2007 and Accepted: March 2, 2008.

[2]: Liu Gang, Wang Xiaochi, "***Improved Bags-of-Words Algorithm for Scene Recognition***", 2012.

[L1]: https://www.tutorialspoint.com/opencv/index.htm consulté la derniere fois le 01/07/2017

[L2]: https://www.tutorialspoint.com/python3/python_overview.htm consulté la derniere fois le 01/07/2017