UNIVERSITÉ DE MONTRÉAL

VERS PLUS DE ROBUSTESSE EN RECONNAISSANCE D'OBJETS ET DE VISAGES POUR L'ANALYSE D'IMAGES ISSUES DE VIDÉOS DE CONCERT

FANNIE PUECH
DÉPARTEMENT DE GÉNIE INFORMATIQUE ET GÉNIE LOGICIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE INFORMATIQUE)
DÉCEMBRE 2012

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

VERS PLUS DE ROBUSTESSE EN RECONNAISSANCE D'OBJETS ET DE VISAGES POUR L'ANALYSE D'IMAGES ISSUES DE VIDÉOS DE CONCERT

présenté par : PUECH Fannie
en vue de l'obtention du diplôme de : Maîtrise ès Sciences Appliquées
a été dûment accepté par le jury d'examen constitué de :

M. MERLO Ettore, Ph.D, président
M. PAL Christopher J., Ph.D membre et directeur de recherche
M. BILODEAU Guillaume-Alexandre, Ph.D, membre

À mes parents.

REMERCIEMENTS

Je remercie tout particulièrement Jean-Nicolas Brunet pour m'avoir écoutée, conseillée, et soutenue dans les moments de doute. Merci également à Samira Ebrahimi et Lucas Berthoux pour leur amitié précieuse. Merci à mon père Denis Puech, ma mère Francoise Puech, et mon frère Pacôme Puech pour leur richesse et leur intelligence. Merci également à mon directeur de recherche Christopher Pal pour son ouverture d'esprit et sa confiance. Merci à Kamrul Hasan pour ses conseils judicieux et son soutien. Merci également au Professeur Merlo, et au Professeur Bilodeau d'avoir accepté d'être membres de mon jury. Merci à la famille Brunet pour son accueil et son soutien. Merci encore à ces quelques professeurs qui au long de mon parcours ont stimulé ma curiosité et mon envie d'apprendre, en particulier Daniel Carrière, Philippe Gaillard et Romuald Debruine.

RÉSUMÉ

Les vidéos de concert constituent un exemple typique de documents très populaires qui sont mal indexés par une description textuelle. Une meilleure indexation passe par l'étude du contenu visuel de ces vidéos. Or, les algorithmes à la pointe en analyse d'images sont encore trop peu robustes au contenu hostile des vidéos de concert. C'est pourquoi, nous nous efforçons ici d'identifier les aspects critiques qui limitent l'efficacité des algorithmes classiques de reconnaissance d'objets et d'individus sur les images complexes. Nous proposons alors, le cas échéant, des pistes de solutions pour rendre ces techniques plus robustes au contenu des vidéos de concert.

Detection d'instruments. Au chapitre un, nous mettons en lumière les facteurs limitant en pratique les performances des méthodes classiques de reconnaissance d'objets appliquées aux vidéos de concert. Pour ce faire, nous révisons l'ensemble du pipeline de détection d'objets à la lumière des contraintes imposées par le contexte de l'analyse vidéo. Nous identifions et décrivons notamment les écueils suivants : la complexité algorithmique des méthodes, la mauvaise gestion de la multinomialité des contenus, et la fragilité des algorithmes face aux images à contenu riche (scènes complexes).

Complexité algorithmique des méthodes. Le goulot d'étranglement du pipeline d'apprentissage en reconnaissance d'objets réside sans conteste dans le calcul du vocabulaire visuel utilisé pour représenter les images sous la forme d'un histogramme de mots visuels. En effet, l'heuristique des k-moyennes est appliqué à l'ensemble des descripteurs locaux extraits des images d'entrainement, soit un ensemble de très grand cardinal et de grande dimension. Le processus est lourd, et de plus, susceptible de converger vers des minimums locaux. Nous proposons ici une méthode de descente avec relance dynamique, qui permet d'éviter un certain nombre de configurations problématiques sans avoir à réinitialiser complètement l'algorithme. Notre méthode constitue une bonne alternative aux algorithmes de recherche locale ou de relance qui ont étés proposés pour pallier aux manquements de l'algorithme des k-moyennes.

Scènes complexes. L'apprentissage sur des bases de données représentatives de la variabilité des images contenues dans les vidéos de concert est un incontournable pour l'obtention d'un classificateur robuste sur un tel contenu. Ainsi, de manière pratique, est-il nécessaire de disposer, dans la base de données d'apprentissage, d'exemples d'instruments présentés dans les mêmes conditions que dans une vidéo de concert (occlusion par le musicien, variation de couleur et de forme). Une telle variabilité peut-être obtenue par collecte automatisée d'images sur le web. Malheureusement, ces images ne sont pas optimisées pour l'apprentissage. Elles consistent en des scènes complexes, incluant l'objet d'intérêt. Or, l'apprentissage

sur des images présentant plusieurs objets dans un environnement complexe n'est pas une tâche triviale. Comme nous le montrons au chapitre 3.2, la présence d'arrière-plan nuit aux performances des algorithmes. Le recours à une boîte englobante pour isoler l'objet sur les images d'entraînement permet de résoudre ce problème, mais suppose une intervention humaine coûteuse. Nous proposons donc une méthode permettant d'estimer automatiquement la position d'un objet donné sur des images d'entraînement.

Multiplicité des classes et multinomialité. Enfin, la conception d'un algorithme adapté à la détection de plusieurs classes d'objets, éventuellement multinomiales, souffre d'un manque d'automatisation. Usuellement, une machine à vecteurs de support linéaire est apprise pour chaque classe d'objets. Nous montrons au chapitre 3.3 que cette pratique courante présente plusieurs limitations. Nous proposons donc une méthode à l'intersection entre le SVM multi-classe et les arbres de décision permettant de gérer un nombre important de classes éventuellement multinomiales. Nous montrons que, à encodage égal, cette méthode permet d'améliorer le F1-score de 10% par rapport à une méthode d'apprentissage standard par SVMs.

Reconnaissance de visages. Dans le second chapitre de ce mémoire, nous évaluons l'état de l'art des techniques de reconnaissance d'individus et leur applicabilité aux vidéos de concert. Nous présentons en particulier la méthode d'apprentissage de métrique pour la comparaison dans l'espace des similarités en cosinus et proposons une amélioration. Nous soulignons ensuite l'impact négatif des grandes variations de la pose des individus et du faible nombre d'images disponibles par personne pour l'apprentissage. Enfin, nous explorons les techniques de classification à grande échelle et les structures de données adaptées.

La *représentation des visages* est différente de la représentation des objets, du fait de leur forme constante. Nous présentons alors les méthodes classiques d'apprentissage et introduisons une méthode de classification basée sur les techniques à la pointe en vérification, en l'occurrence l'apprentissage de métrique pour la similarité en cosinus - CSML. Nous montrons alors que cette technique peut être améliorée par une projection linéaire supplémentaire (LDE) de type Fisher-non paramétrique. L'accroissement de précision observé est de l'ordre de 8%.

La *prise en compte de la pose* en vérification (comparaison deux à deux de visages) permet d'améliorer considérablement les performances des algorithmes. L'idée consiste à apprendre une métrique adaptée pour la comparaison respective des paires de visages de face, de profil, et les paires face-profil. Nous montrons que ce résultat se vérifie aussi en classification. Par ailleurs, le *nombre d'images disponibles* pour l'apprentissage par individu impacte aussi fortement les performances des algorithmes de reconnaissance. Sur la base de données Labelled Faces in the Wild, nous mesurons l'impact du nombre d'instances disponibles pour l'apprentissage sur les performances d'un algorithme de classification.

Enfin, nous examinons le scénario de la *classification rapide de visages à grande échelle*. Nous prouvons la légitimité de la métrique L_2 après projection dans un espace optimisé pour la similarité en cosinus. Nous étudions alors les performances des structures de données hiérarchiques pour la classification rapide sur ces bases de données et montrons que la réduction de dimensionnalité est un incontournable pour l'utilisation efficace de ces structures.

Indexation de vidéos de concert. Enfin, nous proposons une stratégie pratique pour l'indexation des vidéos de concert. La méthode proposée est basée sur une description détaillée du contenu visuel et sur une méthode innovatrice d'indexation des évènements musicaux.

L'*analyse de vidéos* en général est un thème complexe. Ici, la restriction aux vidéos de concert simplifie l'étude puisque l'on sait à quel type de contenu s'attendre. Nous proposons donc de procéder à la mesure de plusieurs indicateurs pour évaluer la qualité d'une vidéo : présence d'instruments et d'individus d'intérêt, qualité de la prise de vue, et popularité du groupe et de l'évènement.

Le *recueil de valeurs* de ces paramètres consiste à mettre en oeuvre les méthodes développées pour la détection d'instruments et de visages. Il s'agit aussi de proposer un moyen fiable permettant d'estimer la qualité de la prise de vue. Nous introduisons donc une méthode basée sur la mesure du flot optique moyen pour estimer les mouvements de caméra. Enfin, nous décrivons une méthodologie simple pour mesurer la popularité d'un groupe ou d'un évènement musical.

Pour finir, fort de ces mesures, nous étudions le problème de l'indexation de vidéos de concert, et proposons une méthode pour la détection de vidéos inappropriées, à savoir, les vidéos ne représentant pas un concert, les vidéos mal filmées, ou encore les vidéos ne présentant pas le groupe de musiciens d'intérêt.

ABSTRACT

While concerts are a popular subject for the videos found online, they are often poorly indexed relative to other types of media. This thesis aims at introducing a strategy to improve concert video indexing using computer vision techniques from the image recognition field. More specifically, we aim at exploring several key aspects of several state of the art techniques that limit their ability to be properly applied to online concert videos.

Instrument detection. In the first chapter, we discuss several factors that bound the performances of traditional object recognition techniques when applied to concert videos. To do so, we go through a commonly used the state of the art object detection pipeline and we identify aspects that are critical for concert video content analysis. More specifically, we identify the following pitfalls : algorithms complexity, poor modeling of content variability, sensitivity to background on complex scenes.

Algorithm complexity. One bottleneck of the learning pipeline in object recognition is the visual vocabulary computation step. Typically, the k-means algorithm is applied to the space of local descriptors extracted from training images, which is a large space populated with dense high dimensional vectors. Hence, we study the complexity of this algorithm and show that a key parameter is the number of centers, k, used for clustering, which also defines the size of the visual vocabulary. We therefore also explore the appropriate size of a visual vocabulary for image recognition algorithms. More specifically, we introduce a reasonable criterion to choose the value of k, given a set of local descriptors. From a more practical point of view, we implemented a heuristic to improve k-means and prevent local minima to be reached too quickly. This algorithm appears to be competitive with state of the art methods for fast k-means computation.

Background clutter. To obtain a robust classifier for images from concert videos, one cannot rely on a small dataset showing objects under an ideal point of view, on centered images without background. To cope with the large variation of shape and point of view of instruments in concert videos, there is no choice but to learn using a representative dataset. Such a dataset can be obtained by mining the web, as it has been done for the well-known image-net database. Still, if representative of the variety of instruments within classes, images from image-net are not provided with bounding boxes. To label images with the location of the object of interest is a costly operation that we cannot consider at large scale. Thus, we introduce an algorithm to estimate an object location within an image by cross-comparing images from the training-set.

Creating efficient, high performance multiclass classifiers To conceive a detection algo-

rithm dealing with several object classes, a standard practice consists of training one SVM per object. We show that this strategy can be improved in several ways, in particular by taking into account class multinomiality and by dealing with several classes at the same time. Specifically, we show that the F-1 score can be improved by more than 10% in comparison to standard one-vs-all SVMs.

Face recognition. In the second chapter, we review face recognition techniques. We focus on Cosine Similarity Learning (CSML) and measure its efficiency when applied to the face recognition problem on faces in the wild from the Labelled Faces in the Wild dataset. We show that CSML can be further improved by Linear Discriminant Embedding. Then, we underline the negative impact of pose variations and of the low number of training images per people. At last, we review large scale classification techniques and appropriate datastructures.

Face representations differ from object because recognition involves exploring details of the face, not an average shape. We introduce standard methods for face comparison and measure the precision one can get by applying these to the classification problem. Specifically, we experiment with the nearest neighbor classification algorithm on faces projected in the CSML space, which is optimized to separate the different faces under cosine similarity. We show that a further projection can improve the classification accuracy by more than 8%.

To explicitly take into account pose on pictures has proven to be a fruitful practise for pairs of faces comparison. The idea consists in learning a specific metric for each pair of pose (frontal-frontal, frontal-side, side-side). Here we show that this observation is also true for the face classification problem. The number of training images per people is another factor that has a major impact on the precision of classification algorithms. On the LFW dataset, we measure the relation between the number of training images and the achieved accuracy. We then discuss the image gathering process and its impact on learning strategies. At last, we discuss the large scale face recognition scenario and we prove that the L_2 metric can be used for nearest neighbor search after CSML on normalized vectors.

Concert video indexing. Finally, we introduce a methodology for concert videos indexing, based on a detailed description of a video visual content and an innovative event indexing strategy.

Video analysis is a wide field of study. Here we focus on concert videos which simplifies the process and allows us to provide a more detailed study. Indeed we know what kind of content one might expect from a good concert video. We thus proceed to measure the following parameters : objects and people of interest, camerawork quality, band popularity and event size.

Features extraction. To gather these features for several videos can be done by using previously developped methods for instruments and person recognition. We then develop

an algorithm to measure the quality of camerawork. We thus introduce a technique based on optical-flow and show that a relationship can be drawn between averaged optical flow per frame and camera movement quality. We also introduce basic methods to evaluate the popularity of a band and the popularity of a musical event.

We conclude by a statistical model that can predict the quality of a concert video using the previously mentioned parameters.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES ANNEXES

LISTE DES PRINCIPAUX SIGLES ET ABRÉVIATIONS

AAM	Model d'Apparence Actif
CS	Similarité en Cosinus
CSML	Apprentissage de Similarité en cosinus
DOG	Différence de Gaussiennes
FW	Faces in the Wild
HOG	Histogramme de Gradients Orientés
LBP	Motif Binaire Local
LFW	Labeled Faces in the Wild
(L)LDA	Ananlyse Discriminante Linéaire (Locale)
(L)LDE	Projection linéaire Discriminante (Locale)
PCA	Ananlyse en Composantes Principales
RBM	Machine de Boltzman Restreinte
RBM	Machine de Boltzman Restreinte
SVM	Machine à Vecteur de Support
SP	Pyramide spatiale

CHAPITRE 1

INTRODUCTION

Si l'on sait aujourd'hui comment l'oeil perçoit le monde -ce qui rend possible la numérisation du visuel- on ignore encore comment le cerveau l'interprète. La recherche pour la compréhension automatisée des images s'intéresse à ce processus flou de construction de sens à partir de données numériques primitives. À la manière du cerveau traitant les simuli visuels reçus par l'oeil, les algorithmes de reconnaissance d'image essaient de faire émerger une interprétation de haut niveau de la représentation exhaustive, mais brute des images numériques.

Tandis que le langage parlé se construit à partir d'un alphabet, de mots, de phrases, de paragraphes, selon des règles établies, l'image, elle, est dépourvue de cette hiérarchie de standards qui facilitent la transmission de sens. De plus, si les frontières entre concepts sont bien définies pour le langage, dans le domaine visuel, on observe une divergence entre les unités cohérentes par le sens, ou la désignation parlée, et les unités cohérentes par l'aspect. Ces caractéristiques du monde visuel expliquent sans doute le retard des méthodes d'indexation d'images et de vidéos en ligne sur les méthodes d'indexation de textes.

Les vidéos en ligne, pourtant au coeur de l'économie numérique, sont particulièrement mal indexées du fait de l'écart entre l'information visuelle et le texte associé. Le recours à des techniques d'analyse d'image est une étape incontournable pour une meilleure gestion de ce type de contenu. Malheureusement, sur des images issues de vidéos, la variabilité des formes et points de vue, la diversité des expressions et des poses des individus, font que les méthodes à la pointe en traitement d'images peinent encore à réaliser une analyse de précision. Les vidéos de concert, très populaires, constituent un exemple de contenu multimédia critique pour les moteurs de recherche. Si des mécanismes rétroactifs permettent d'évaluer l'intérêt des internautes, ils sont sujets aux singularités et ne sont efficaces qu'à long terme. Aussi, existe-t-il une demande prégnante, concrète, pour un système capable de décrire avec précision le contenu de telles vidéos.

Dans le cadre de ce mémoire, nous étudions les facteurs limitant les performances des algorithmes de reconnaissance d'objets et d'individus appliqués aux vidéos (chapitres 3 et 4). Nous proposons également une méthodologie efficace pour l'indexation par le contenu de vidéos de concerts (chapitre 5), basée sur les outils présentés aux chapitres précédents.

Figure 1.1 La trahison des images, Renée Magritte

1.1 Définitions et concepts de base

1.1.1 Représentation des images

Le nombre d'images utilisées pour l'apprentissage (et donc la représentativité des données d'entraînement) est un facteur critique pour la qualité des algorithmes de reconnaissance d'objets et d'individus. Or, l'espace de stockage des données est limité et la complexité des algorithmes d'apprentissage dépend fortement de la taille des objets à manipuler. La représentation des images est donc soumise à deux enjeux contradictoires : la préservation de l'information et le besoin de compacité en mémoire.

Descripteurs locaux

Une image en couleur de hauteur h et de largeur w peut être représentée par une matrice 3D de dimension $w * h * d$. La troisième dimension correspond au nombre de vecteurs constituant la base de représentation des couleurs ($d = 3$ sauf exception). En reconnaissance d'objets, on néglige souvent cette dimension car la couleur ne constitue pas un invariant de la description d'une classe d'objets. On représente donc une image par une matrice I de dimension $h * w$, en niveaux de gris.

Figure 1.2 Un objet n'est pas caractérisé par sa couleur (Andy Warhol, Cows, 1971-76)

Même sans la dimension de couleur, une représentation exhaustive est lourde pour des images de définition standard. De plus, une comparaison pixel à pixel est très sensible aux

changements d'échelle, de couleur et de forme. Torralba, Fergus et Freeman proposent une classification basée sur la représentation matricielle d'images de très basse résolution pour la classification de scènes (Torralba *et al.*, 2008a), mais leur approche reste exceptionnelle. Aujourd'hui, afin d'alléger la représentation, on utilise généralement une approche basée sur une étude des gradients (Scale Invariant Feature Transform -SIFT, Speeded Up Robust Features -SURF, Histograms of Oriented Gradients -HOG) ou la détection de formes caractéristiques (lignes, angles saillants) via la convolution avec des noyaux spécifiques. Ces techniques permettent de sous-échantillonner la matrice des images pour ne conserver que l'information primordiale. Dans le cadre de ce mémoire, nous utilisons des descripteurs SIFTs pour la reconnaissance d'objets (Chapitre 3) car ils sont reconnus comme de bons invariants aux rotations, translations et variations d'échelle. Par ailleurs, ils constituent un standard en terme de description d'images.

Le principe des descripteurs SIFTs a été introduit par Lowe à la fin des années quatre-vingt-dix (Lowe, 1999). Il s'agit de décrire localement les points clés d'une image par des histogrammes de gradient. Ces histogrammes sont normalisés ce qui rend la représentation invariante aux changements d'échelle tandis qu'un alignement sur le gradient dominant permet de rendre la représentation invariante aux rotations.

Parmi les stratégies de segmentation de l'image en un ensemble de points clés, on compte notamment l'échantillonnage régulier, aléatoire, la détection des invariants ou encore la détection de schémas de gradient caractéristiques. L'implémentation (Hess, 2010) utilisée dans le cadre de ce mémoire est basée sur la détection de points correspondants aux extrêma dans une pyramide de différences de gaussiennes (Difference Of Gaussians -DOG). Une k-pyramide de gaussiennes est obtenue en appliquant à une image k convolutions avec des filtres gaussiens de variance croissante en k (l'image devient donc de plus en plus floue). Une pyramide de différences de gaussiennes résulte de la soustraction deux à deux des étages consécutifs de la pyramide obtenue.

Cette procédure de sélection de points clés dans l'image s'accompagne usuellement d'un filtrage destiné à éliminer les points situés sur des motifs à courbure trop faible (pseudolignes), trop fréquents d'image en image et peu informatifs quant au contenu.

Pour la reconnaissance de visages, nous utilisons une description plus dense appelée Motif Binaires Locaux (Local Binary Patterns -LBPs). Comme les SIFTs, les LBPs sont des descripteurs locaux des variations d'intensité. Leur construction est différente de celles des SIFTs cependant. Selon l'idée originale de Ojala et al. (Ojala *et al.*, 1994), le voisinage immédiat d'un pixel (fenêtre 3x3) est décrit par un nombre binaire. Chaque digit de ce nombre correspond à la différence d'intensité entre un pixel de la fenêtre et celle de son voisin (1 si l'intensité du pixel est supérieure à celle de son voisin, 0 sinon). Etant plus dense que la

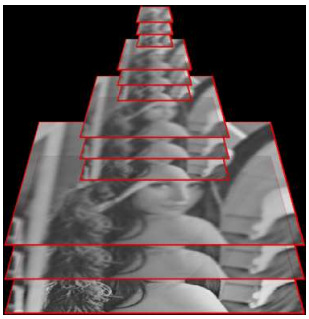

Figure 1.3 Pyramide de gaussiennes

representation par SIFTs, les LBPs sont plus adaptées aux objets texturés à étudier dans le détail, tels que les visages.

Mots visuels

La représentation des images sous forme d'histogrammes de descripteurs locaux est une stratégie introduite par des chercheurs du Xerox en 2004 (Csurka *et al.*, 2004). Dans leur article, Csurka et al. intronisent la notion de mots visuels - ou *bag of words*, qui permet de construire une description des images extrêmement compacte. Aujourd'hui, cette représentation est très largement utilisée par la communauté scientifique et dans le monde industriel.

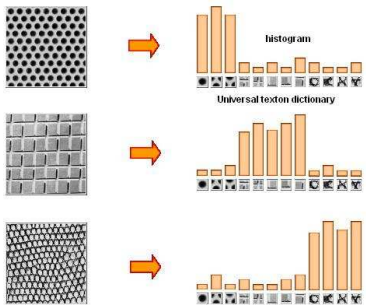

Figure 1.4 Principe de l'encodage par sacs de mots visuels

La quantification par sacs de mots visuels, consiste à condenser une image sous la forme d'un histogramme d'index, relatifs à un corpus de mots visuels. Les mots visuels sont simplement des descripteurs représentatifs de la distribution des descripteurs locaux extraits sur les images d'entraînement. Chaque descripteur d'une image se voit associer l'index de son voisin dans ce lexique, ce qui conduit à la formation d'un histogramme de mots visuels représentant

l'image. Ainsi, une image est representée par un vecteur de m-mots-visuels dimensions, plutot que par n vecteurs de m' dimensions ($m' = 128$ pour les descripteurs SIFTs, $n \approx 2000$).

1.1.2 Apprentissage

Des finalités de la reconnaissance d'images

En reconnaissance d'objets, on distingue la tâche de reconnaissance d'une instance spécifique d'objet (l'auto cabossée de mon frère) de la tâche de reconnaissance d'une classe d'objets en général (les autos). Dans le second cas de figure, il est nécessaire d'apprendre une représentation suffisamment généralisante pour résister aux variations d'aspect (mon auto est moins cabossée que celle de mon frère, plus spacieuse, d'un joli rose - la couleur, les bosses, la taille de l'auto doivent être exclues de la représentation).

Figure 1.5 Divergeance concept-aspect dans le domaine du visuel (MOMA, design over time)

En reconnaissance de classes d'objets, on distingue les objets à contours finis (auto, guitare, arbre), les concepts plus vagues tels que les textures (eau, ciel, herbe, sable, bois), les scènes (cirque, ville, campagne), ou même les idées plus abstraites telles que les actions (jouer de la guitare, manger, courir) ou les sentiments (heureux, apeuré). Les visages humains sont traités à part, comme c'est le cas dans le cerveau humain. Leur morphologie particulière induit de la régularité dans les données dont on peut tirer partie pour la comparaison des images. Dans le cadre de ce mémoire, nous adressons essentiellement le thème des objets à contours finis et des visages.

Algorithmes d'apprentissage et évaluation des performances

Disposant d'une banque d'images encodées, $X = (I_1, ..., I_n)$, assortie d'un étiquetage $C = y_1, ..y_c$, le travail du chercheur en intelligence artificielle consiste à apprendre une fonction f capable de caractériser les concepts, ou classes, de C présents sur les image de X.

La classification restreinte consiste à apprendre f telle que :

$$f(I_i \in X) \in \{y_1, ..y_c\}, \tag{1.1}$$

La classification binaire consiste à apprendre f telle que :

$$f(I_i \in X) \in (\{0,1\}, .., \{0,1\}_c), \tag{1.2}$$

L'évaluation d'une procédure d'apprentissage est réalisée au moyen d'indicateurs mathématiques au nombre desquels figurent notamment la précision et le rappel *(precision, recall)*. Suivant que l'on considère la tâche de classification binaire ou restreinte, le mode de calcul de ces indicateurs de performance varie.

Considérons un problème à trois classes A,B,C. À partir d'un jeu de test, on peut construire une matrice de confusion associée au problème (Tableau 1.1). Le nombre d'exemplaires du jeu de tests appartenant à la classe A, classés correctement, est noté TP_A (pour *True Positive* A). De même pour TP_B et TP_C. Le nombre d'exemplaires de A classés B est lui noté e_{ab} (e pour erreur de a vers b). De même pour e_{ac}, e_{ba}, e_{bc} et ainsi de suite.

En classification restreinte, pour ce problème, on a :

$$FP_a = e_{ba} + e_{ca},\ FN_a = e_{ab} + e_{ac} \tag{1.3}$$

Alors, la précision (pre) et le rappel (rap) sont respectivement donnés par :

$$prec_A = \frac{TP_A}{TP_A + FP_A},\ rap_A = \frac{TP_A}{TP_A + FN_A} \tag{1.4}$$

Tableau 1.1 Matrice de confusion en classification

classe	A	B	C
A	TP_A	e_{AB}	e_{AC}
B	e_{BA}	TP_B	e_{BC}
C	e_{CA}	e_{CB}	TP_C

En classification binaire :

TP = vrais positifs, soit les détections réalisées à bon escient (True Positive).
FP = faux positifs, soient les détections positives à mauvais escient (False Positive).
TN = vrais négatifs, pas de détection et rien à détecter (True Negative).
FN = faux négatifs, pas de détection malgré la présence de l'objet (False Negative).
De même qu'en classification restreinte, on a alors :

$$prec = \frac{TP}{TP + TN} \text{ et } rap = \frac{TP}{TP + FN} \tag{1.5}$$

Stratégies d'apprentissage

Étant donné (X,C), on peut envisager deux approches pour apprendre f capable de caractériser les classes C de X : l'approche discriminative ou l'approche générative. La première stratégie, l'approche discriminative, consiste à apprendre directement la distribution des étiquettes sachant les données $P(C|X)$ (on estimera par exemple l'équation d'une frontière entre les classes). La régression logistique est un exemple de ce type de stratégie. La seconde approche, l'approche générative, consiste à apprendre à modéliser chaque classe de C et à réaliser la classification d'un exemplaire non étiquetté en testant son adéquation avec les modèles appris. En termes probabilistes, on apprend donc à modéliser les données sachant l'étiquette, ou classe, soit $P(X|Y)$. En utilisant la règle de Bayes, on peut alors estimer :

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)} \text{ [1]}. \tag{1.6}$$

Une approche discriminative : les machines à vecteurs de support

En vision par ordinateur, les méthodes à la pointe tant dans la sphère de la recherche que dans le monde industriel utilisent massivement une méthode discriminative appelée machines à vecteurs de support (SVM).

Les machines à vecteurs de support sont des classificateurs binaires, appartenant à la classe des séparateurs. étant donné $X = (x_1, ..x_n)$ vecteurs d'entraînement et $Y = (y_1, ..y_n)$ étiquettes, S un ensemble d'entrainement $S = \{(x_i, y_j)\}$, S est dit séparable si $\exists f$ telle que :

$$\forall (x_i, y_j),\ f(x_i, y_j) > 0 \text{ ssi } S(x_i, y_j) = 1,\ < 0\ sinon. \tag{1.7}$$

f est appelé séparateur. Si f est linéaire, on dit que les données sont linéairement séparables.

La particularité qui fait des SVMs des séparateurs exceptionnels réside dans le fait que les SVMs sont des séparateurs à marge maximale. La marge d'un séparateur γ_f est une grandeur qualifiant la distance des exemples à l'hyperplan du séparateur. Pour un exemple donné x, la marge de f en x, $\gamma_{f,x}$ est la norme du vecteur orthogonal à l'hyperplan défini par f passant par x. On a alors :

$$\gamma_f = min_x(\gamma_{f,x}). \tag{1.8}$$

1. En pratique, le dénominateur, $P(X)$, n'a pas à être calculé car : $arg\ max_y(\frac{P(X|Y)P(Y)}{P(X)}) = arg\ max_y(P(X|Y)P(Y))$.

La forme de f varie suivant que les données sont ou non linéairement séparables. Dans le cas positif, f est de la forme :

$$f(x) = x'w + b, \tag{1.9}$$

avec b l'origine de l'hyperplan et w une droite directrice de l'hyperplan séparateur.

L'équation de l'hyperplan peut être obtenue en résolvant un problème d'optimisation suivant :

$$min\frac{1}{2}w.w', \text{ sc } \forall j,\ wx_j + b >= 1. \tag{1.10}$$

Ou en introduisant une pénalité lagrangienne :

$$min\frac{1}{2}w.w' - \sum_j \alpha_j((wx_j + b)y_j - 1). \tag{1.11}$$

avec $\alpha_j > 0 \forall j$.

On a : $w = \sum_i \alpha_i y_i * x_i$ et $b = y_k - wx_k$, pour k quelconque avec $\alpha_k > 0$, d'où la formulation duale :

$$L_D = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j x_i x_j \tag{1.12}$$

Dans le cas non linéaire, on utilise une projection ϕ dans un espace de dimension supérieure où les données sont linéairement séparables. La construction de l'hyperplan séparateur dans cet espace suit la même méthodologie que dans le cas linéaire, modulo une fonction objectif légèrement modifiée :

$$L_D = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j k(x_i, x_j) \tag{1.13}$$

En notant que le problème de calcul de l'hyperplan séparateur à marge marginale ne fait appel qu'au produit scalaire des vecteurs et non explicitement aux vecteurs de l'espace, on peut alors avoir recours à l'astuce du noyau consistant à ne pas calculer explicitement $\phi(x), x \in X$ mais seulement $k(x, y) = \phi(x)\phi(y)$, avec ϕ la projection dans un espace de plus grande dimension.

L'inconvénient des SVMs réside dans leur caractère binaire. S'il existe plusieurs formulations multiclasses de l'algorithme (Hsu et Lin, 2002), l'apprentissage reste basé sur la combinaison de SVMs binaires de sorte que les calculs deviennent rapidement lourds lorsque le nombre de classes augmente.

Classificateur par le plus proche voisin

Dans le cadre de ce mémoire, nous nous intéressons à un type particulier de classificateur appelé classificateur par le plus proche voisin. Ce classificateur présente d'intéressantes propriétés en présence d'un nombre élevé d'instances d'apprentissage.

Sous sa forme élémentaire, la classification par la recherche du plus proche voisin est triviale et ne requiert pas d'apprentissage. On parle de méthode non paramétrique. étant donné x de label inconnu, x est classé comme appartenant à la classe de son voisin dans l'ensemble d'entraînement muni d'une distance appropriée. La stratégie de classification par le plus proche voisin repose sur l'intuition suivante : si l'on possède suffisamment d'exemples de chaque classe, c'est à dire, si l'espace d'entraînement est suffisamment dense, le plus proche voisin d'une instance inconnue constitue un bon classificateur. L'intuition est bonne, puisque Bengio et Bottou montrent (Bottou et Bengio, 1995) que si le nombre d'exemples tend vers l'infini, l'erreur théorique est bornée par le double de l'erreur de Bayes, qui est l'erreur commise minimale étant donné l'ensemble d'apprentissages.

L'ensemble d'apprentissage n'atteignant jamais la perfection théorique (classes qui se chevauchent, mauvaise couverture de l'espace), le classificateur par le plus proche voisin est susceptible de commettre des prédictions erronées. Toutefois, des méthodes d'apprentissage (génératives ou discriminatives) savamment intégrées au modèle permettent de réduire les erreurs commises et de faire du classificateur par le plus proche voisin un classificateur compétitif avec l'état de l'art. De plus lorsque les données sont fortement multinomiales et donc difficilement séparables, le classificateur par le plus proche voisin prend en compte naturellement la topologie des données, à l'inverse des méthodes paramétriques classiques.

Sous sa version approchée, le classificateur par le plus proche voisin (Approximated Nearest Neighbor), permet de tirer facilement partie d'ensembles d'entraînement de très grande taille. En effet, si la complexité d'une recherche exacte du plus proche voisin croît linéairement avec n, le nombre d'instances d'apprentissage, des structures de données hiérarchiques de type arbre à couverture (cover-tree) ou arbres de métriques (metric trees), permettent de réaliser une recherche en temps sous linéaire en contrôlant la précision, là ou les algorithmes exacts échouent du fait de leur complexité.

Ainsi, à l'échelle du web, le classificateur par le plus proche voisin a-t-il sans doute un rôle clé à jouer. Dans le cadre de ce mémoire, nous nous attachons à montrer que cet algorithme permet d'obtenir des résultats de l'ordre de l'état de l'art en terme de précision tout en offrant des possibilités de mise à l'échelle dépassant celles offertes par les autres classificateurs.

1.1.3 Indexation d'images et de vidéos

Dans son sens le plus large, l'indexation consiste à associer à un document plusieurs concepts clés. Ces concepts peuvent être des mots du langage courant, mais aussi, pour les images, des notions plus abstraites ou plus proches de la sémantique visuelle. En recherche d'images et de vidéos en ligne, on distingue l'indexation par le texte et l'indexation par le contenu. L'indexation par le contenu consiste à décrire une image ou une vidéo par l'analyse des données visuelles brutes, et non par le texte environnant le document. Même si l'association de méta-données normalisées aux documents multimédias est en passe de devenir systématique, pour des raisons de compacité, les descriptions réalisées restent très succinctes. De plus, elles reposent sur la bonne foi de l'auteur du document. De surcroît, la description humaine d'une image est systématiquement subjective, interprétative, là où l'indexation par le contenu produit des éléments d'information impartiaux et normalisés. Pour réaliser une bonne indexation de documents visuels, il est donc nécessaire d'avoir recours à des méthodes de compréhension de l'image.

Figure 1.6 Le problème d'indexation de vidéos de concert, nécessité de l'analyse du contenu (Source : YouTube).

1.2 Éléments de la problématique

La place croissante du support multimédia en ligne (e-TV, YouTube) rend très concret le besoin d'un système automatisé capable d'analyser le contenu des vidéos du web. Les applications d'un tel système s'étendent de l'indexation au contrôle du contenu (droits d'auteur, pornographie), en passant par le placement de produits et le profilage utilisateur.

Malgré les progrès considérables observés depuis une dizaine d'années en analyse de l'image[2], l'indexation de vidéos en ligne reste un problème ouvert en 2012. Le contenu de

2. La précision obtenue sur la base de données Caltech101 est passée d'environs 20% en 2000 a plus de

vidéos constitue un défi pour la recherche du fait de plusieurs facteurs critiques, au nombre desquels figurent notamment l'échelle des problèmes, la diversité des formes au sein des classes d'objets et de la pose au sein des classes de visages, ainsi que la richesse du contenu des images.

Dans le cadre de ce mémoire, nous abordons ces trois notions, d'abord du point de vue de la reconnaissance d'objets (chapitre 3), puis, dans un second temps, du point de vue de la reconnaissance de visages (chapitre 4). Au travers de ces deux chapitres, nous discutons des défis de la recherche face à la réalité des vidéos de concert, dans le scénario d'une application pratique à l'échelle du web. Nous proposons des pistes de solution pour une plus grande robustesse à ce type de contenu. Fort de ces méthodes, nous discutons de la mise en place d'un système d'indexation et d'analyse de vidéos d'évènements musicaux (chapitre 5) basé sur l'analyse et la compréhension des images.

1.2.1 Techniques de reconnaissance d'objets adaptées aux vidéos de concert

Première branche de la recherche en traitement d'image abordée dans ce mémoire, la reconnaissance d'objets est un thème qui occupe les chercheurs depuis près de quarante ans. Aujourd'hui, les travaux semblent converger vers une méthodologie globale en trois étapes, à savoir la description locale du contenu de l'image, l'encodage de l'information au moyen de sacs de mots visuels, et l'apprentissage d'un classificateur (souvent une machine à vecteur de support).

Description locale des images et dimensionnalité de la représentation

Si les méthodes de description locale gèrent haut la main les changements d'échelle, les rotations et les occlusions, elles posent néanmoins le problème de la dimensionnalité de la représentation. Encodée au moyen de descripteurs locaux, une image I est représentée par un ensemble d'environ mille descripteurs de plus de cent dimensions, soit plus de cent-mille coefficients réels. La dimension importante de la représentation a naturellement guidé les recherches vers des méthodes de réduction de la dimensionnalité. Aujourd'hui, la stratégie de compression sous forme de sacs de mots visuels est admise comme un standard par la communauté scientifique. Se pose alors la question de la taille du corpus de descripteurs de référence, qui conditionne la dimensionnalité de la représentation des images. Fixée de façon empirique aux alentours de mille mots pour la plupart des applications industrielles et de recherche, la taille du vocabulaire visuel est un thème rarement abordé par les chercheurs sous l'angle théorique. En pratique, ce paramètre est ajusté en fonction de la taille de la banque

70% en 2010

de données traitée. Il conditionne directement l'efficacité des processus d'apprentissage et d'analyse des images.

La taille du vocabulaire visuel agit comme un curseur sur le degré de spécialisation de la représentation. En effet, plus la taille est importante, plus les distinctions entre les descripteurs du corpus de mots visuels sont subtiles et moins la représentation est généralisante. Toutefois, à l'inverse de ce que suggère l'intuition, accroître le vocabulaire visuel ne permet pas nécessairement d'obtenir une meilleure classification. Ainsi, une trop grande discrimination entre les mots visuels dissimule les points communs entre les images d'apprentissage appartenant à la même classe. En termes plus techniques, la qualité de la description de la distribution des descripteurs par un corpus de mots visuels n'est pas nécessairement linéairement liée a la taille du corpus.

Quel(s) critère(s) adopter pour choisir la dimensionnalité de la représentation des images ? Dans le cadre d'une banque d'images à grande échelle, quels algorithmes permettent de construire un vocabulaire visuel en un temps raisonnable ? Au prix de quelles approximations ? Ces questions sont abordées au paragraphe 3.1. du présent document.

Encodage d'images à contenu complexe

Bien que reconnue comme une technique incontournable en traitement d'image, la méthode d'encodage par sacs de mots sous sa forme originelle, telle qu'introduite dans l'article du Xerox de 2004, souffre de plusieurs manquements.

D'abord, la représentation *binaire* des mots visuels a été remise en cause pour la drastique perte d'information qu'elle engendre et les ambiguïtés qu'elle génère. Si la perte d'information inhérente à la quantification peut être réduite en augmentant conséquemment la taille du corpus visuel utilisé pour l'encodage, cette stratégie impacte fortement la complexité de la procédure d'apprentissage via l'augmentation de la dimensionnalité de la représentation des images. Or, le but premier de la quantification sous forme de mots visuels est de réduire la dimensionnalité de la représentation des données. Il est donc légitime de se demander si un vocabulaire de dimension moindre ne doit pas être préféré, quitte à avoir recours à des techniques d'encodage plus complexes que l'encodage binaire élémentaire.

Ensuite, la représentation par sacs de mots visuels est très sensible à la présence d'arrière-plan sur une image. Dans ce contexte, les descripteurs appartenant à l'objet d'intérêt sont susceptibles d'être noyés par les descripteurs de l'arrière-plan. Or, dans le scénario qui nous intéresse, les images d'entraînement sont imparfaites et l'arrière-plan occupe une place importante. Si des méthodes pour la soustraction d'arrière-plan existent, elles présentent une complexité non applicable dans le présent contexte. Une alternative consiste alors à pratiquer un apprentissage par zone, mais ceci suppose d'avoir recours à un processus lourd d'étiquetage

manuel des images consistant à localiser l'objet d'intérêt sur une image complexe. *Quelles stratégies d'encodage sont-elles envisageables à grande échelle ? Comment se passer du processus d'étiquetage manuel dans le scénario d'un apprentissage sur des images complexes ?* Cette problématique est traitée au paragraphe 3.2.

Classification

L'étape de classification consiste à apprendre une fonction capable d'associer les étiquettes adéquates à une image inconnue. Sur de grandes bases de données, il est nécessaire de porter attention à la complexité de la phase d'apprentissage comme de la phase de classification à proprement parler. La méthode discriminative de classification par SVM a remporté de brillants succès en reconnaissance d'objets depuis les années 2000. Cependant, elle souffre de l'accroissement du nombre de classes d'objets à traiter. En effet, classifier une image inconnue au moyen de SVMs revient à tester un à un les SVMs appris pour chacune des classes d'intérêt. La SVM multiclasse n'apporte qu'une réponse partielle à cette problématique, puisqu'elle ne consiste qu'à apprendre les paramètres permettant de trancher entre plusieurs SVMs binaires. Par ailleurs, lorsque le nombre de classes augmente ou que les variations inter-classe se font plus importantes, les noyaux linéaires des SVMs échouent à séparer correctement les données. Or, l'entrainement de SVMs à noyaux non linéaires n'est pas envisageable à grande échelle. La classification par SVM est-elle adaptée au type de problème que nous traitons ?

Pour les très grands jeux de données, le recours à une structure arborescente devient rapidement incontournable. Celles-ci sont naturellement associées à la technique de classification par la recherche approchée du plus proche voisin. Cependant, la classification par le plus proche voisin souffre d'imprécision du fait de la finitude du nombre d'exemples d'apprentissage et du phénomène de granularité observé au niveau de la frontière entre les classes. *Quelles sont alors les performances de l'algorithme sur de grands jeux de données ? Quelles méthodes permettent de renforcer le classificateur par le plus proche voisin ? Comment intégrer les machines à vecteurs de support à un classificateur hiérarchique ?*. Ces questions sont l'objet du chapitre 3.3.

1.2.2 Techniques de reconnaissance de visages adaptées aux vidéos de concert

Le cerveau traite différemment les objets en général et les visages humains. Du point de vue de la recherche en vision par ordinateur, la reconnaissance d'individus par l'analyse des traits du visage diffère de la reconnaissance d'objet en ce qu'on essaie de construire une représentation capturant non pas les caractéristiques générales de l'objet visage, mais les traits spécifiques d'un individu. Les méthodes utilisées pour répondre à ce problème sont

donc très différentes. Toutefois, le domaine de reconnaissance de visage est soumis aux mêmes contraintes d'échelle (nombre de classes), et de variabilité inter-classe (variation dans la pose). à ces défis s'ajoute celui de la disponibilité des exemples d'apprentissage, du fait des limitations dues à la législation sur la vie privée des personnes.

Modélisation du visage humain et stratégies de classification

Le fait que l'on se spécialise dans l'étude d'un type d'objet particulier, le visage, dont les caractéristiques morphologiques sont connues, permet de mettre en oeuvre des techniques d'alignement et de mise à l'échelle non envisageables pour les objets. Ainsi, en détectant des points caractéristiques tels que les yeux, les coins du nez, la bouche, on peut procéder à la comparaison de deux images sur une échelle commune. La précision de ces opérations a un impact conséquent sur les performances du classificateur.

La réduction de la dimensionnalité des images pour la comparaison de visages diffère de celles des objets. En effet, pour les visages, il est pertinent d'utiliser une approche de type Analyse en Composantes Principales(PCA) puisque les composantes importantes pour la comparaison des individus sont grossièrement constantes. Cette observation suggère le recours à la classification par la comparaison d'images, c'est à dire, par la recherche d'une image semblable au visage inconnu dans un jeu de données d'entraînement. Depuis une dizaine d'années, les travaux sur l'apprentissage de distance pour la comparaison des visages ont mis en évidence les bonnes performances de la similarité en cosinus. étant donné un jeu de paires d'images représentant un même individu ou deux individus différents, il est aujourd'hui aisé d'apprendre une fonction optimisant la séparation dans l'espace des scores de similarités des paires correctes et incorrectes. Ce processus est appelé vérification. La précision moyenne en vérification atteint les 90% sur la base de données Labeled Faces in the Wild.

Malheureusement, être capable de distinguer des paires d'individus identiques et des paires d'individus différents n'implique pas d'être capable de reconnaître l'identité d'un inconnu dans une vidéo. L'utilisation d'un vérificateur dans ce contexte implique de comparer le visage inconnu à une plétorie de visages et est susceptible de retourner plusieurs identités. *Comment trancher ? Quelle stratégie adopter pour pratiquer la classification de visages en tirant partie des acquis de la vérification ? Dans quelles mesures les performances obtenues en vérification sont-elles dégradées ?* Nous traitons cette question au chapitre 4.1.

Pallier à la limitation du nombre d'exemples d'apprentissage et aux variations de la pose

Récemment, il a été mis en évidence qu'une distinction selon la pose des individus contribuait à améliorer l'apprentissage de distance en cosinus. Cette remarque est particulièrement pertinente dans le cadre de notre travail puisque les individus apparaissant dans des vidéos de concert sont en mouvement constant. Ainsi, il est important de disposer d'un classificateur robuste pour les visages vus de face, mais aussi vus de profil. Or, si l'on comprend que l'apprentissage d'une métrique revient (grossièrement) à sélectionner les pixels pertinents pour la comparaison de deux individus, il est naturel de songer qu'une distinction selon la pose serait profitable. De plus, du fait de la quantité limitée d'images d'apprentissage disponibles, il est nécessaire de tirer profit de chaque exemplaire. Aussi, souhaite-t-on maximiser la comparaison visage vue-de-profil et visage vue-de-face par exemple.

Un autre procédé susceptible de renforcer les performances d'un classificateur consiste à rassembler un nombre suffisant d'images d'apprentissage en pratiquant l'apprentissage semi-supervisé. étant donné un certain nombre d'images étiquetées avec certitude, des exemplaires supplémentaires peuvent être collectés et filtrés au moyen d'un vérificateur. *Comment un classificateur de visage à grande échelle supporte-t-il le scénario où peu d'images d'entraînement sont disponibles pour chaque individu ? Dans quelle mesure un apprentissage spécifique à la pose renforce-t-il un classificateur ? Comment se procurer des images supplémentaires ?* Des réponses à ces questions sont apportées au paragraphe 4.2.

Similarités et métriques pour la classification à grande échelle

Les résultats obtenus en vérification pour le problème de reconnaissance de visage suggèrent le recours au classificateur par le plus proche voisin. Par exemple, on pourra procéder à la recherche du plus proche voisin d'un visage inconnu dans une banque de visages étiquetés, assortie d'une vérification. La complexité de la recherche brute du plus proche voisin est en $O(n * d)$, avec n le nombre de visages candidats à la comparaison, et d la dimension de la représentation (environs deux-cents dimensions pour les visages). Aussi, est-il indispensable d'avoir recours à une structure arborescente dédiée à l'optimisation du procédé. On pourra par exemple avoir recours aux kd-arbres (*kd-trees*), aux arbres à couverture (*cover trees*), ou aux arbres étendus (*spill-trees* (Liu *et al.*, 2004c)). Lorsque la dimension des vecteurs est importante, les deux dernières structures sont réputées plus efficaces.

Se pose alors la question de la métrique. Le processus d'apprentissage d'une similarité en cosinus (Cosine Similarity Metric Learning) a démontré son efficacité sur de nombreux jeux de données. Malheureusement, la similarité en cosinus ne vérifie pas les axiomes de base

d'une distance. Or, les structures de données adaptées à la recherche rapide du plus proche voisin exploitent massivement la propriété de l'inégalité triangulaire. *Dans quelle mesure les structures hiérarchiques permettent-elles d'accélérer la classification ? Jusqu'à quelle dimension ? Peut-on justifier l'usage de la distance euclidienne dans l'espace des comparaisons en cosinus ? Ne peut-on améliorer l'optimisation réalisée par cet apprentissage ?* Ces questions sont l'objet du paragraphe 4.3.

1.2.3 Étude de cas : évaluation de la qualité des vidéos de concert en ligne

L'indexation de vidéos est un domaine de la recherche particulièrement dynamique. Le volume de contenu vidéo en ligne augmente rapidement et le besoin d'un système d'indexation automatisé devient chaque jour plus prégnant. Les techniques utilisées pour l'indexation dépendent largement de l'application. En effet, le mode d'archivage des vidéos ainsi que la finalité de l'outil à concevoir conditionnent les choix stratégiques.

Ici, nous nous attachons à concevoir un système permettant d'indexer et de décrire le contenu des vidéos de qualité représentant un concert. Nous avons opté pour ce type de vidéos, car elles sont particulièrement populaires et très mal indexées [3]. De plus, elles sont représentatives d'un type de vidéo -les vidéos évènementielles- pour lesquelles le besoin d'un système automatisé d'indexation est particulièrement important.

Analyse de vidéo

L'analyse de vidéo en général est un processus complexe, car le contenu est non restreint. Le problème que nous posons ici est plus spécifique, car nous adressons uniquement les vidéos de concert. Ainsi, peut-on profiter du fait que l'on sait sous quelle forme se présente le contenu recherché et pratiquer une analyse plus fine que celle réalisée pour indexer une vidéo quelconque. La qualité des images, la qualité de la prise de vue et du son, le lieu, le groupe musical et la date du concert impactent bien entendu la popularité d'une vidéo musicale. De même, la définition des images et le contenu à proprement parler ont sans conteste un impact sur la qualité perçue d'une vidéo. *Quels paramètres est-il raisonnable d'essayer d'estimer ? Quel est leur ordre d'importance ? Quelles sont les classes d'objets à définir ?* Nous abordons ces questions au chapitre 5.1.

3. Résultat de la recherche *ACDC+live+HD* : http ://www.youtube.com/watch ?v=JV7I3Z–gDo, 8ème position

Extraction de l'information

Traiter l'aspect évènementiel. En terme d'indexation de vidéos de concert, une première question à se poser consiste à se demander s'il est possible de mesurer la popularité d'un groupe musical et d'un évènement. Wikipédia permet d'obtenir une liste de groupes par genre musical. Fort de cette liste, il est possible d'obtenir une liste de concerts pour ces artistes au moyen d'un bot minant les sites de vente de billets en ligne [4]. De cette façon, un nombre conséquent de dates et lieux de concerts peut être collecté. *Comment évaluer la popularité de ces évènements ? Comment évaluer la popularité d'un groupe du musicien ?* Cette question est traitée au paragraphe 5.2.

Filtrage de la qualité. L'analyse de la qualité de la prise de vue peut être réalisée selon plusieurs indicateurs. Ici, nous nous proposons d'utiliser la taille des visages détectés, comme le suggère la littérature. Toutefois, cette indication semble insuffisante, car elle ne prend pas en compte les mouvements de caméra (phénomènes de tremblements, mouvements brusques). *Comment mesurer la qualité des mouvements lors de la prise de vue ?* Une réponse à cette problématique est apportée au paragraphe 5.2.

Analyse du contenu. Disposant de vidéos de concert, malgré le filtrage par la qualité de l'image et la nature de l'évènement, il est nécessaire de procéder à une analyse permettant de déterminer ce qu'il est réellement donné de voir. En effet, une vidéo de bonne qualité, associée par son titre à un évènement populaire, peut néanmoins se révéler hors sujet. Aussi, proposons-nous de détecter des instruments clés et des visages via les méthodes robustes développées aux chapitres précédents. *Quelle est l'efficacité de ces méthodes ?* Cette question est également traitée au paragraphe 5.2.

évaluation de la qualité d'une vidéo de concert

étant donné les paramètres énumérés jusqu'ici, il reste à déterminer le poids de ces facteurs sur la qualité d'une vidéo musicale. Nous proposons donc de pratiquer une analyse statistique élémentaire pour déterminer la contribution respective de chacun de ces paramètres sur la popularité des vidéos en ligne. Pour ce faire, nous nous référons aux indicateurs mesurant les consultations de la vidéo et l'évaluation de la vidéo via des votes binaires (bon, mauvais contenu). *Quels paramètres impactent fondamentalement la popularité d'une vidéo musicale ? Peut-on prédire la popularité d'une vidéo au moyen de ces caractéristiques ? Quel modèle statistique ou d'apprentissage permet-il d'estimer la popularité d'une vidéo de concert ?* Nous abordons ce questionnement au chapitre 5.3 de ce mémoire.

4. example : http ://www.stubhub.com, http ://www.songkick.com

1.3 Objectifs de recherche

La problématique développée ci-avant nous amène à définir la démarche décrite ci-après.

Obj1 : Détection d'instruments. évaluer les défis rencontrés par les algorithmes de reconnaissance d'objets à l'épreuve des vidéos de concert et proposer des solutions adaptées le cas échéant.

Obj1.1 : Mettre en évidence un critère théorique permettant de justifier le choix de la dimension de la représentation des images et proposer une heuristique adaptée à la construction de vocabulaires visuels sur des bases de données de grande taille.

Obj1.2 : étudier les différentes stratégies d'encodage au moyen d'un vocabulaire visuel et mettre en évidence la problématique des images avec arrière-plan. Proposer des solutions au problème d'achalandage d'histogramme dû à la présence d'arrière-plan complexe.

Obj1.3 : évaluer les performances d'un classificateur par le plus proche voisin en présence d'un nombre suffisant d'instances et proposer, le cas échéant, des solutions pour en améliorer la robustesse.

Obj2 : Reconnaissance d'individus. évaluer les défis rencontrés par les algorithmes de reconnaissance d'individus à l'épreuve des vidéos de concert et proposer des solutions adaptées le cas échéant.

Obj2.1 : étudier les modes d'encodage et de comparaison des visages et proposer une stratégie de classification de visages tirant partie des stratégies de vérification existantes. Proposer le cas échéant des pistes d'amélioration.

Obj2.2 : Proposer une stratégie de classification tenant compte explicitement de la pose des individus et discuter de la collecte automatisée d'exemplaires supplémentaires pour l'apprentissage.

Obj2.3 : étudier le passage à la très grande échelle et les structures ou stratégies adaptées à la classification rapide par le plus proche voisin.

Obj3 : Concevoir un système pour l'indexation et l'analyse de vidéo de concert de qualité.

Obj3.1 : étudier les méthodes classiques de classification de contenu vidéo et identifier les spécificités inhérentes à l'analyse de vidéos de concert.

Obj3.2 : Proposer des méthodes pour l'analyse du contenu des vidéos de concert, mesurant respectivement la qualité de l'image, la popularité de l'évènement associé et la présence d'objets ou d'individus d'intérêt.

Obj3.3 : Proposer une méthode d'évaluation de la qualité des vidéos de concert et ligne.

1.4 Plan du mémoire

Le présent mémoire s'articule en trois chapitres thématiques et une revue de littérature. Le chapitre 2 donne un apperçu global des articles ayant trait au thème de la reconnaissance d'images et de visages à adapter au scénario des vidéos de concert. Le chapitre 3 répond aux objectifs Obj1.* en étudiant les algorithmes de reconnaissance d'objets adaptés à la détection d'instruments dans des vidéos musicales. Le chapitre 4 aborde ensuite les objectifs Obj2.* concernant les visages. Enfin, le chapitre 5 traite les objectifs Obj3.* en propose une méthodologie pour l'évaluation automatisée de la qualité des vidéos de concert.

CHAPITRE 2

REVUE DE LITTÉRATURE

Comme nous l'avons souligné dans la problématique, la conception d'algorithmes robustes pour l'analyse de vidéos de concerts suppose de relever trois défis, à savoir l'échelle des problèmes, la richesse du contenu des images, et la variabilité interne des classes d'objets et de visages.

2.1 Reconnaissance d'objets

Dans ce mémoire, nous prenons le parti d'entraîner un détecteur d'instruments en nous référant à une base de données d'images représentative de la diversité des formes et points de vue observés dans les vidéos de concert. En pratique, cela implique de travailler avec plus de cinq-cents images par classe d'objets. Aussi, sommes-nous amenés à employer des méthodes adaptées aux problèmes à grande échelle [1].

2.1.1 Approches paramétriques adaptées à la grande échelle

Au nombre des approches paramétriques appliquées à la reconnaissance d'image à grande échelle, on compte notamment le *boosting*, les réseaux de neurones, les machines à vecteur de support, et les forêts de décision randomisée.

Boosting. Malgré sa simplicité, le *boosting* a souvent été utilisé pour la classification d'images. Pour mémoire, cette technique d'apprentissage consiste à construire un classificateur complexe constitué d'une série de classificateurs élémentaires (*weak classifiers*) qui, judicieusement combinés, constituent un classificateur robuste. Plus précisément, étant donné un jeu d'exemplaires d'apprentissages X, un jeu de labels associés Y, et un jeu de poids W, il s'agit d'apprendre itérativement un classificateur élémentaire associé au triplet (X,Y,W). Les poids sont initialement égaux et mis à jour à chaque itération de sorte que les exemplaires mal classifiés à l'itération t sont plus lourds à l'itération t+1. La simplicité de la procédure d'apprentissage fait de l'algorithme de *boosting* un bon candidat pour la classification à grande échelle. En 1999, Viola et al. utilisent cette stratégie pour pratiquer la recherche d'images dans une base de données d'environs 3,000 images (Tieu et Viola, 1999). Plus tard, Torralba

1. En reconnaissance d'images, on parle de problème à grande échelle passé le millier d'images. Ici, nous travaillons avec 3500 images pour la base de données d'instruments issue d'image-net, plus de 5000 images pour PASCAL VOC

et al. utilisent le *boosting* pour la recherche d'instances d'objets (Torralba *et al.*, 2008b) sur une base de données à très grande échelle de l'ordre du million d'images.

Réseaux de neurones. Dans le même article de Torralba et al., le *boosting* est comparé à une stratégie d'apprentissage basée sur une Machine de Boltzman Restreinte (Restricted Boltzman Machine). Les Machines de Boltzman Restreintes sont des réseaux de neurones sans connexion directe au sein d'une même couche de neurones. L'apprentissage est réalisé par descente de gradient selon la dérivée de la fonction d'énergie du réseau. L'apprentissage d'une RBM est réputé complexe, cependant, les RBMs ont récemment été appliquées avec succès à des bases de données de grande dimension. En 2008, Torralba propose d'utiliser les RBMs sur une base de données de très grande taille (Torralba *et al.*, 2008b) via le recours à un encodage binaire extrêmement compact (256 bits par image). Avec un encodage plus standard, les RBMs sont également appliquées à la reconnaissance d'objet par Norouzi en 2009 (Norouzi *et al.*, 2009). La méthode proposée par Norouzi et al. constitue l'état de l'art pour la reconnaissance de piétons sur la base de données dédiée de Caltech [2] (250,000 images).

Forêts randomisées. à l'inverse des structures de données hiérarchiques non paramétriques classiques, les arbres de décision procèdent à des choix informés (basés sur l'étiquetage des données) pour construire des structures hiérarchiques de décision. Plus spécifiquement, la scission des noeuds est réalisée en conscience des étiquettes associées aux données (Quellec *et al.*, 2010) via une heuristique bien choisie, de manière à séparer au mieux les classes contenues dans un noeud. Les forêts de décision randomisées ont été créées pour pâlier au problème de sur apprentissage observé avec les arbres de décision. Les arbres de décision étant fortement basés sur les données d'apprentissage, ils sont en effet sujets à ce type de comportement. Depuis les années 2000, les forêts de décision randomisées (Random Decision Forest) ont été massivement appliquées au problème de classification à grande échelle (B. Frohlich (2011), H. Fu (2012), Quellec *et al.* (2010), Criminisi *et al.* (2012)). En 2011, Frohlich utilise une RDF sur la base de données NUS-WIDE, qui comporte 261,000 images représentant 81 concepts. Il utilise un processus de décision gaussien dans chaque feuille des arbres de la forêt pour réaliser la classification finale (B. Frohlich, 2011), et renforce ainsi la précision du classificateur construit. Plus récemment, Fu, Zhan, et Qiu ont proposé un modèle de RDF pour l'annotation d'images à grande échelle (H. Fu, 2012). La stratégie pour la scission des noeuds consiste à réduire la dimension des vecteurs au moyen d'une analyse par composantes principales et à générer plusieurs échantillons aléatoires sur les données réduites pour choisir la meilleure scission possible. La procédure proposée constitue l'état de l'art en 2012 sur la base de données Corel5k, qui comporte 5,000 images.

SVMs. Enfin, les machines à vecteur de support constituent la technique de référence en

2. http ://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/

classification d'images. à grande échelle, les SVMs (linéaires) ont notamment été appliqués à image-net, et constituent depuis 2010, l'état de l'art sur cette base de données, avec un taux de classifications correctes de 53%. Les performances de cette méthode découlent de l'utilisation d'un vocabulaire de grande taille, de plusieurs types de descripteurs, et d'un encodage par pyramide spatiale. De même, les SVMs constituent l'état de l'art sur la base de données Pascal du VOC. La méthode la plus performante recensée sur cette base de données est basée sur l'approche de quantification par sacs de mots visuels et le recours aux SVMs non linéaires (Chen *et al.*, 2010). Les auteurs de cette méthode soulignent que les performances de leur algorithme découlent de leur stratégie particulière de construction de pyramide spatiale. En 2012, dans un tutoriel, Perronnin et al. dressent un panorama des méthodes appliquées avec succès au problème de classification à grande échelle (Perronnin *et al.*, 2012). Ils préconisent les méthodes basées sur les SVMs linéaires, associées à des vecteurs de grande dimension, et entraînées au moyen d'une descente de gradient stochastique.

2.1.2 Approches non paramétriques adaptées à la grande échelle

Plus proche voisin. La principale méthode non paramétrique appliquée à la classification d'images à grande échelle est la classification par le plus proche voisin. Bien que cette technique ait été longtemps considérée comme une méthode imprécise, des voix se sont récemment élevées pour souligner qu'elle ne requiert pas d'entraînement, souffre peu de sur apprentissage, et permet de supporter les bases de données à très grande échelle. En particulier, en 2008, Boiman et al. soutiennent que la quantification des images au moyen de sacs de mots visuels nuit à ce classificateur et que le retour à une classification basée sur les descripteurs (SIFT) permet d'obtenir des résultats compétitifs avec l'état de l'art en utilisant la recherche du plus proche voisin (Boiman *et al.*, 2008). Dans l'article de Boiman et al., la recherche approchée du plus proche voisin est réalisee au moyen d'un kd-arbre sur l'ensemble X des descripteurs d'entraînement. Le résultat de la classification c d'une image I, correspond alors au minimum de la distance cumulée par classe de chacun des descripteurs de I à leur voisin dans X.

Les résultats énoncés par Boiman et al. sont repris et confortés par Huynh et al. en 2010 (Jacobs et Huynh, 2010).

2.1.3 Approches hybrides

Dans leur article, Boiman et al. ne s'attardent pas sur la distance définie sur l'espace de recherche. Or, il s'agit d'un facteur impactant profondément les performances du classificateur par le plus proche voisin. Le classificateur par le plus proche voisin est fréquemment évalué sur

des espaces métriques munis d'une distance élémentaire. Optimiser cette distance, localement ou globalement ôte certes son caractère non paramétrique à la méthode, mais permet d'affiner conséquemment la classification.

L'apprentissage d'une distance pour le classificateur par le plus proche voisin est un thème largement exploré en traitement d'images. Le calcul de distance peut être global ou local, réalisé dynamiquement lors de la classification ou calculé au préalable. Les méthodes globales formulent fréquemment le problème d'apprentissage de distance comme un problème d'optimisation, consistant à séparer les paires d'instances similaires des paires d'instances différentes. Si on note + l'ensemble des paires d'images d'une même classe, − l'ensemble des paires de classes différentes, une formulation élémentaire du problème est alors :

$$\begin{gathered} min_A \sum_{(i,j)\in+} \|x_i - x_j\|_A \text{ , sc,} \\ A > 0 \text{ et } \textstyle\sum_{(i,j)\in-} \|x_i - x_j\|_A > 1 \end{gathered} \tag{2.1}$$

Cette idée est à l'origine d'approches telles que l'Analyse Discriminante Linéaire (*Linear Discriminant Analysis*), et la Projection Linéaire Discriminante (*Linear Discriminant Embeding*). Ces techniques ont été massivement appliquées en reconnaissance de visages. En ce qui concerne les objets, la préférence est allée aux méthodes locales, telles que la Pertinence Locale des Composantes - LFR (Ho *et al.*, 2008), l'apprentissage adaptatif de noyau (Domeniconi *et al.*, 2002), l'apprentissage local par SVM et les versions locales de la LDA et de la LDE (*Local LDA* et *Local LDE* Chen *et al.* (2005)). Chaque méthode correspond à une variante du problème de classification énoncé plus haut, appliqué localement. En 2006, Zhang et al. utilisent l'apprentissage local sur la base de données Caltech-101 et obtiennent l'état de l'art sur cette base de données (Zhang *et al.*, 2006).

Enfin, il reste à mentionner les méthodes d'analyse par composantes - NCA (Goldberger *et al.*, 2004), et d'analyse de pertinence des composantes -RCA (Tsang *et al.*, 2005). La RCA consiste à apprendre une distance basée sur la matrice de covariance locale des éléments d'une classe. La NCA quant à elle maximise le score obtenu dans une configuration de validation croisée de type *leave-one-out*. En 2004, Goldberg et al. appliquent la NCA à la base de données USPS (Goldberger *et al.*, 2004), supplantant les approches par PCA et par LDA précédemment appliquées sur cette base de données.

2.1.4 Apprentissage par région

L'apprentissage par région consiste à sélectionner sur les images d'entraînement des zones discriminantes pour la classification (Deng *et al.* (2009),Bangpeng et Li (2010)). Ce type

de procédure est motivé d'une part par le problème d'achalandage d'histogrammes, d'autre part par la nécessité d'identifier des détails discriminants pour la classification (Yao *et al.*, 2011a) lorsque la frontière entre les classes est très fine (lorsqu'il s'agit de distinguer une guitare classique d'une guitare électrique par exemple, on ne peut se contenter des cordes). Ces deux problématiques ont récemment suscité un interêt particulier, car elles constituent des jalons incontournables pour l'application des algorithmes de reconnaissance d'objets au contenu multimédia du web.

Le problème de l'arrière-plan est abordé par Yakhnenko et al., en 2011. Ils segmentent les images d'entraînement selon une grille régulière et entrainent un classificateur discriminant de type SVM sur un ensemble de ces régions étiquetées *arriére-plan* ou *objet d'intérêt* (Yakhnenko *et al.*, 2011). Récemment Yao et al. ont utilisé une approche consistant à construire une forêt randomisée d'arbres de décision pour identifier les régions discriminantes pour la classification fine entre classes d'objets semblables (Yao *et al.*, 2011a). En 2011, cette méthode constitue l'état de l'art sur la base de données Caltech-UCSD-birds, qui contient des images de 200 espèces d'oiseaux.

2.1.5 Bases de données, ordres de grandeur

La base de données Pascal du VOC possède 20 classes d'objets comptant de 150 a 1,400 images d'entraînement. L'état de l'art en 2011 est détenu par Tsang et al. avec une méthode basée sur l'apprentissage de SVMs non linéaires (Tsang *et al.*, 2005) après encodage sous forme d'histogrammes. L'originalité de la méthode réside dans le mode particulier d'encodage des descripteurs locaux au moyen d'un vocabulaire visuel de moyenne dimension. Sur la base de données image-net-10k, constituée de 10,000 classes d'objets issues d'image-net, la méthode la plus performante en 2011 repose sur l'entraînement de SVMs linéaires employées à classifier des images encodées sous forme de vecteurs de Fisher. Plusieurs types de descripteurs sont utilisés pour la classification, recourant aux textures, aux couleurs, et aux gradients locaux (SIFT).

Tableau 2.1 Bases de données et résultats de référence en classification

Base de données	état de l'art	Méthode
Pascal-2011	56.5%-95% (AP)	Chen *et al.* (2010)
imageNet-2010	53% (AP)	linear-SVM-VQ-SPM

2.2 Classification d'individus

2.2.1 Reconnaissance de visages de sujets non conditionnés

Nature et disponibilité des images. La comparaison de visages est longtemps restée cantonnée à des portraits de sujets conditionnés, c'est à dire, des images centrées sur des visages posant de face, sans variations d'intensité majeures ou de jeux de lumière complexes. Afin de construire un système capable d'adresser la diversité des visages extraits de vidéos de concert, il est nécessaire de sortir de ce cadre idéaliste et de considérer la réalité des visages extraits de vidéos. La base de données Labelled Faces in the Wild (LFW) a été conçue pour représenter le contenu du web. Ainsi, les photographies de LFW présentent des visages sans restriction sur la pose ou les conditions de lumière. Par ailleurs, le nombre d'images disponibles pour l'apprentissage d'un classificateur varie d'individu en individu, selon une courbe en exponentielle décroissante. La problématique de la disponibilité d'images d'entraînement en reconnaissance de visages est notamment mentionnée par Stone et al.(Stone *et al.*, 2010). Posant le problème de l'apprentissage d'un classificateur d'individus, ils soulignent que le nombre de photographies moyen disponibles en ligne pour une personne est inférieur à cinq photographies.

Alignement. L'alignement est un incontournable en classification de visages. La procédure consiste à détecter des points clé sur les visages pour les aligner sur un axe commun de l'espace en 3D. Les méthodes basées sur l'AAP (Active Appearance Model) sont très largement utilisées dans la littérature (Valstar *et al.* (2010), Vukadinovic et Pantic (2005)) du fait de leur grande précision. Ces méthodes présentent cependant l'inconvénient d'être très lourdes d'un point de vue computationnel (le temps de traitement est de l'ordre de la minute par image). En 2011, Hasan et Pal présentent un dispositif d'alignement capable de traiter les images en quasi-temps réel, basée sur la détection et la mise en correspondance de régions du visage (M. K. Hasan, 2011).

L'écart dans la pose des individus est sans conteste un facteur important nuisant à la comparaison des visages. Il est donc naturel d'avoir l'intuition d'une procédure distinguant les cas selon la pose. La prise en compte de la pose des individus pour l'alignement a été notamment traitée par Odobez et al. puis par Dong et al. En 2009. Odobez et al. proposent une procédure capable de différencier 91 poses différentes du visage (Ricci et Odobez, 2009). Dong et al. s'intéressent plus tard a 21 poses (Dong *et al.*, 2010). En 2012, Hasan et al. soulignent l'avantage d'une distinction selon 3 poses sur la procédure d'apprentissage (Hasan *et al.*, 2012) en reconnaissance de visages.

2.2.2 Comparaison d'images

Sur la base de données LFW, les performances des algorithmes sont comparées selon leur aptitude à différentier les paires de visages identiques des paires de visages composées de deux identités distinctes (on parle de vérification). Plusieurs scénarios sont alors proposés pour l'évaluation des algorithmes : (a) l'expérimentation est conduite sans données externes, avec les seules images fournies par LFW, (b) l'expérimentation a recours à des données extérieures. Dans le cadre de ce mémoire, nous nous plaçons dans le premier cas de figure (a). En 2009, Wolf et al. obtiennent un taux de vérifications correctes de 86% pour (a) (Wolf *et al.*, 2009a). Ils utilisent, après une phase d'alignement, une description au moyen de plusieurs descripteurs : LBP, TPLBP, FPLBP et SIFT. La classification est basée sur deux extensions de la technique d'analyse locale discriminante de Fisher (LDA), l'OSS (One Shot Similarity measure) et la TSS (Two Shots Similarity measure). Dans le cadre du second scénario (b), l'état de l'art est représenté par Yin et al., dont la méthode est basée sur le recours à une base de visages externe permettant d'évaluer les variations de l'apparence d'un individu et d'affiner la prédiction finale (Yin *et al.*, 2011). Le score obtenu en vérification sous cette configuration dépasse les 90%. La stratégie d'apprentissage de similarité en cosinus (Cosine Similarity Metric Learning) a prouvé sa robustesse en terme de comparaison de visages à travers de nombreux articles de la littérature. Introduite par Nguyen et al. en 2010, la méthode consiste à apprendre une mesure de similarité optimisée pour la comparaison des visages (Nguyen et Bai, 2010). L'avantage de cette méthode est qu'elle s'accompagne d'une réduction importante de la dimensionnalité, ce qui permet une classification rapide. Une revue complète des résultats obtenus sur LFW est donnée par Huang et al. (Huang *et al.*, 2005).

2.2.3 Classification d'individus à grande échelle

Le problème de la reconnaissance de visages est rarement formulé en termes de classification (jusqu'à présent nous avons évoqué la tâche de vérification). En 2011, Rim et Pal procèdent à la classification des 50 identités correspondant aux individus possédant le plus d'images dans LFW. Ils atteignent une précision de 82% en ajoutant des données bruitées issues de vidéos YouTube (Rim et Pal, 2011). En 2009, Guillomin et al. rapportent des résultats concernant la classification de dix-sept sujets (Guillaumin *et al.*, 2009). En 2011, Wolf et al. s'intéressent à un ensemble plus large de 610 individus. La classification est réalisée au moyen d'une SVM multiclasse et la précision atteinte est de l'ordre de 25%. En ajoutant des données additionnelles, ils atteignent un taux de 45% de précision pour 100 sujets (Wolf *et al.*, 2011).

2.2.4 Bases de données, ordre de grandeur

Les résultats donnés ci-après sont à prendre avec parcimonie. En effet, les performances dépendent très largement du nombre de visages disponibles pour l'entraînement. Ainsi, Rim et al. travaillent sur une base de données dont chaque individu possède au moins 20 visages, tandis que Wolf et al. utilisent beaucoup moins d'images d'entraînement, ce qui impacte bien sûr négativement les performances.

Tableau 2.2 Bases de données et résultats de référence en reconnaissance de visages

Base de donnee	Tâche	Nombre d'individus	Etat de l'art	Méthode
LFW	Verification	-	90.5% AP	Yin *et al.* (2011)
LFW	Classification	50	82% prec.	Rim et Pal (2011)
LFW	Classification	100	24% prec.	Wolf *et al.* (2011)

2.3 Recherche rapide du plus proche voisin

La recherche exacte et linéaire du plus proche voisin dans un espace X de cardinal n et de dimension k peut être réalisée en $O(n * k)$ opérations élémentaires. L'accélération de la recherche du plus proche voisin via l'utilisation de structures de données adaptées est un sujet de recherche qui occupe les chercheurs depuis des décennies. En 2004, Liu et al. dressent un panorama des techniques de recherche approchée du plus proche voisin applicables en reconnaissance d'images (Liu *et al.*, 2004a). Nous invitons le lecteur intéressé à consulter ce document pour une revue exhaustive des techniques de recherche du plus proche voisin à grande échelle. Ici, nous présentons trois structures de données classiques pour la recherche rapide du plus proche voisin dans un espace de grande dimension.

2.3.1 kd-arbres

Les kd-arbres sont des arbres binaires conçus pour la recherche rapide du plus proche voisin dans un espace à k dimensions, avec $k > 1$. Le principe de construction d'un kd-arbre repose sur la partition successive de l'espace selon un hyperplan orthogonal à l'une des dimensions de l'espace dont l'équation est obtenue via une heuristique élémentaire (point médiant). La complexité associée à la construction de l'arbre est en $O(n * log(n))$, car chacun des n points de X doit être placé dans l'arbre selon une procédure en $O(log(n))$. La complexité de la procédure de recherche exacte du plus proche voisin dépend, elle, de la distribution des données sous la distance définie sur X. En 1977, Lee et Wong (Lee et Wong, 1977) montrent que la complexité au pire des cas de la recherche du plus proche voisin dans un

kd-arbre équilibré est en $O(kn^{1-\frac{1}{k}})$. On constate donc que la dimension a un impact majeur sur la complexité de la procédure. Pour la recherche exacte du plus proche voisin, les kd-arbres sont appropriés pour les jeux de données de petite dimension ($k < 10$). En 1999, Maneewongvatana et Mount étudient la recherche approchée du plus proche voisin dans les kd-arbres (Maneewongvatana et Mount, 1999). Selon l'article, la complexité au pire des cas pour une $(1 + \epsilon)$-approximation du plus proche voisin est en $O(log(n) + \frac{1}{\epsilon^k})$. On note là encore l'impact négatif de la dimension. Une analyse complète du comportement des kd-tree est réalisée par A. W. Moore (Moore, 1991).

2.3.2 Arbres de métriques

Les arbres métriques (vp-arbres (Kunze et Weske (2010), Yianilos (1993)), M-arbres (Houten *et al.*, 2005)) sont plus adaptés aux espaces de grande dimension que les kd-arbres car ils partitionnent l'espace en hypersphères centrées sur un ou des points bien choisis, de sorte qu'ils sont moins dépendants de la dimension des vecteurs de l'espace.

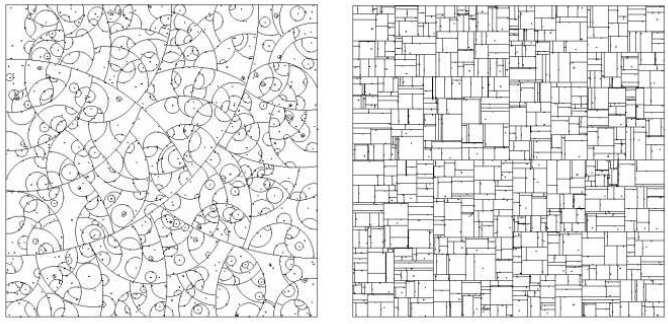

Figure 2.1 Arbre métrique (gauche) versus kd-arbre (droite)

L'évaluation de la complexité théorique de la recherche approchée du plus proche voisin dans un arbre métrique est plus complexe que pour les kd-arbres et n'est pas abordée ici. En pratique, ces arbres sont réputés retrouver exactement le plus proche voisin d'une requête plus rapidement qu'un kd-arbre lorsque la dimension est élevée. Ces structures sont, comme leur nom l'indique, basées sur la notion de métrique en particulier sur l'hypothèse que l'inégalité triangulaire est vérifiée par les éléments de l'espace. Cette dernière propriété permet en effet d'écarter certaines branches de l'arbre lors de la recherche du plus proche voisin, et donc, de minimiser le nombre de calculs de distance réalisés sur l'espace. Nous reviendrons sur cette question au chapitre 4.3.

Comme dans les kd-arbres, la recherche exacte du plus proche voisin dans un arbre métrique repose sur un mécanisme de *backtrack*. Il s'agit de s'assurer que toutes les branches pouvant potentiellement contenir le voisin de la requête aient bien été explorées. Les arbres étendus (*spill-trees*) sont une variante des arbres métriques permettant de minimiser le poids de la procédure de *backtrack*. L'idée essentielle des *spill-trees* consiste à rendre perméable la frontière entre les enfants d'un noeud. Ainsi, les points situés dans a une distance $d < \tau$ de la frontière sont dupliqués dans les deux enfants du noeud. En procédant de la sorte, on peut garantir une recherche approchée du plus proche voisin dont l'erreur est bornée et dépend de τ. Plus τ est élevé, plus le plus proche voisin retrouvé dans la première feuille est proche du plus proche voisin exact. En pratique, τ doit être borné sous peine d'entrer dans un processus de construction infini. Les arbres étendus hybrides (Liu *et al.*, 2004b) alternent entre des opérateurs de scission perméables et imperméables, procurant un bon compromis précision complexité pour une recherche défaitiste (dans la première feuille trouvée).

2.3.3 Arbres à couverture ou *cover-trees*

L'arbre à couverture, ou cover tree est une structure de données hiérarchique complexe (Beygelzimer *et al.*, 2006) conçue pour les espaces de grande dimension dont la constante d'expansion est limitée. La complexité des opérations de routine pour un arbre à couverture dépend des propriétés de la métrique définie sur l'espace. Ainsi, la construction est en $O(c^6 nlog(n))$ et la recherche exacte du plus proche voisin, en $O(c^{12}log(n))$, avec c la constante d'expansion de la métrique sur l'espace (Clarkson, 1997). c est définie comme le plus petit c tel que :

$$\forall p \in X, \forall r \geq 0, |B_X(p, 2r)| \leq c|B_X(p, r)| \tag{2.2}$$

Le cover tree s'est expérimentalement révélé propre à accélérer la recherche rapide du plus proche voisin sur de nombreuses bases de données d'images, dont IRIS et mnist.

CHAPITRE 3

TECHNIQUES DE RECONNAISSANCE D'OBJETS ADAPTEES AUX VIDEOS DE CONCERT

Aujourd'hui, la plupart des algorithmes de reconnaissance d'objets suivent une méthodologie en trois étapes : réduction de la dimensionnalité, encodage sous forme d'histogrammes de mots visuels, et apprentissage d'un classificateur. Notre réflexion s'articule donc naturellement autour de ces trois jalons, que nous examinons en terme d'échelle, d'adaptation à la variabilité des contenus, et de prise en compte de l'arrière-plan des images. Ces trois aspects conditionnent en effet la robustesse des méthodes de reconnaissance d'objets appliquées aux vidéos de concert, et plus largement, aux vidéos complexes. Dans une démarche linéaire guidée par la méthodologie standard en reconnaissance d'objet, nous examinons donc les méthodes à la pointe envisageables dans le cadre du scénario qui nous intéresse et proposons, le cas échéant, des pistes d'amélioration.

Réduction de la dimensionnalité. Dans la première partie de ce chapitre, nous explorons les stratégies de réduction de la dimensionnalité pour l'encodage des images. La constitution du vocabulaire visuel est une étape incontournable qui conditionne la dimension de la représentation des images en réduisant l'ensemble des descripteurs locaux d'entraînement à un ensemble de référence de petite taille. La complexité de ce processus de réduction dépendant du nombre de descripteurs d'entraînement, nous décrivons des algorithmes rapides pour la constitution d'un vocabulaire visuel à grande échelle. Ce faisant, nous soulignons certains points clés du processus et proposons des pistes d'amélioration. D'un point de vue plus théorique, nous nous interrogeons : quelle est la taille appropriée pour vocabulaire visuel en classification d'objets ? Dans quelle mesure accroître la taille du vocabulaire visuel permet-il d'améliorer la description d'une base de données d'images ? Jusqu'à quel point l'augmentation de la dimensionnalité de la représentation des images est-elle raisonnable ?

Encodage. Dans la seconde partie, nous étudions les méthodes d'encodage des descripteurs au moyen du vocabulaire visuel. Nous examinons les différentes stratégies d'encodage (binaire, relatif), et étudions leur complexité. Dans un second temps, nous traitons le cas critique des images avec arrière-plan (*background clutter*). Nous décrivons les limites du processus traditionnel d'encodage sous forme d'histogramme lorsque les images d'entraînement consistent en des scènes complexes. Nous demandons alors : dans la mesure où le nombre d'images sans arrière-plan est limité et incapable de représenter la variabilité intraclasse nécessaire à l'apprentissage pour le contenu vidéo, comment tirer profit des images amateurs

du Web pour l'entraînement des algorithmes ?

Classification. Enfin, dans la troisième partie, nous présentons des méthodes adaptées à la classification à grande échelle. Nous explorons en particulier les paramètres clés qui conditionnent les performances d'un classificateur par le plus proche voisin, que nous comparons au standard des SVMs linéaires. Nous étudions les limites respectives de ces algorithmes et proposons une méthode originale en réponse aux questions suivantes : comment adapter les SVMs à des bases de données multinomiales à grande échelle ? Comment rendre plus robuste un classificateur par le plus proche voisin sans pénaliser outre mesure la complexité de la phase d'entraînement et de classification ?

3.1 Réduction de la dimensionnalité

3.1.1 Position du problème

L'encodage sous forme de sacs de mots visuels implique de représenter la distribution des s descripteurs SIFTs extraits sur les images d'entraînement par $k << s$ vecteurs de même dimension qui sont représentatifs de la distribution. Ces k vecteurs sont appelés mots visuels. Chaque descripteur SIFT extrait sur une image d'entraînement peut alors être indexé par son voisin dans le corpus visuel constitué. Dans le cadre de la méthodologie standard en reconnaissance d'objets, la procédure utilisée pour le calcul de k vecteurs visuels est non supervisée.

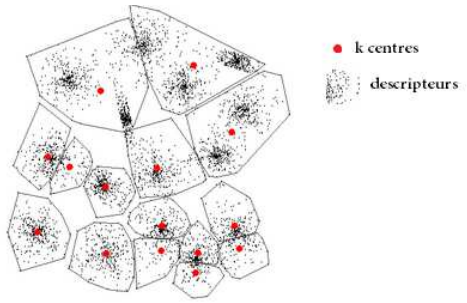

Figure 3.1 Réduction de la dimensionnalité à l'aide de l'algorithme des k-moyennes (2D)

Formellement, étant donné un espace $X = \{x_1, ...x_n\}$ constitué de n vecteurs (i.e. descripteurs) de dimension d, k étant le nombre de clusters (i.e de mots visuels) à constituer, il s'agit d'identifier la position de k centres $(cl_1, .., cl_k)$ dans $\mathbb{R}^d$ qui minimisent la somme des

erreurs de distorsion au carré (Residual Sum of Squared (RSS) errors) :

$$f_{RSS}(cl_1, ..cl_k) = \sum_{j\in[1..k]} \sum_{x_i \in Cl_j} \|x_i - cl_j\|^2 \tag{3.1}$$

L'ensemble Cl_j correspond à la cellule de Voronoi relative au centre cl_j, c'est à dire à l'ensemble des vecteurs de X tel que leur plus proche voisin parmi les centres $(cl_1, .., cl_k)$ est le centre cl_j.

La RSS est une fonction de k qui tend vers 0 quand k tend vers n (un point de X par centre). La RSS ne permet donc d'optimiser une configuration de centres $(cl_1, ..., cl_k)$ que pour k fixé. Pour rendre l'optimisation indépendante de k, et ainsi, optimiser également la taille du vocabulaire, il est nécessaire d'ajouter à la fonction objectif 3.1 un paramètre pénalisant la complexité du modèle (soit le nombre de clusters k), par exemple :

$$f'_{RSS} = f_{RSS}(cl_1, ..cl_k) + \lambda k. \tag{3.2}$$

Le Critère d'Information d'Akaike (AIC) (équation (Drineas *et al.*, 1974)), qui mesure la cohérence d'un modèle avec une distribution statistique, permet de formaliser cette idée. Ainsi, selon l'AIC, le nombre optimal k_{opt} de centres est donné par :

$$k_{opt} = min_k(-2L(k) + 2Q(k)). \tag{3.3}$$

avec L(k) la log-probabilité maximale du modèle à k clusters, et Q(k) le nombre de paramètres indépendants du modèle.

Appliqué, à l'algorithme des k-moyennes, l'AIC 3.3 est formulé comme :

$$k_{opt} = min_k(RSS(cl_1, ..cl_k) + 2dk), \tag{3.4}$$

avec d la dimentionnalité des vecteurs de l'espace de recherche. Soit ici :

$$k_{opt} = min_k(RSS(cl_1, ..cl_k) + 256k) \tag{3.5}$$

En pratique, l'AIC repose sur des hypothèses fortes (indépendance) qui ne sont que partiellement vérifiées par une distribution de descripteurs SIFTs. Ce critère formel est donc peu commode. Aussi, en pratique, k est-il déterminé empiriquement.

Par la suite et jusqu'à nouvel ordre, la norme utilisée est la norme euclidienne. Dans ce contexte, pour k-fixé, le problème de k-clustering optimal est NP-complet, même avec $k = 2$ (Drineas *et al.*, 2004). Toutefois, des heuristiques permettent d'approximer l'optimum en un temps raisonnable. La pratique la plus courante consiste à utiliser l'heuristique des

k-moyennes (expliquée plus bas) et à pratiquer le clustering non supervisé sur la distribution des descripteurs locaux extraits des images d'entraînement. Cette stratégie non supervisée constitue un bon compromis entre les stratégies de quantification supervisées (complexes), et la stratégie consistant à choisir simplement une base d'encodage aléatoire. Parmi les stratégies non supervisées, l'algorithme des k-moyenne apparait comme l'un des meilleurs algorithmes de réduction de la dimensionnalité pour ce problème (Liu *et al.*, 2004c).

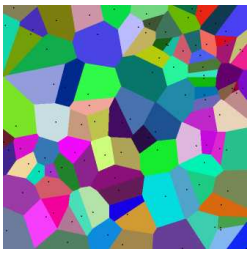

Figure 3.2 Cellules de Voronoi (2D)

3.1.2 L'heuristique des k-moyennes

En matière de clustering non supervisé, l'algorithme des k-moyennes est sans conteste l'heuristique la plus utilisée. Elle vise à minimiser la fonction objectif définie plus haut (équation 3.1). Nous rappelons ci-après les grandes lignes de cet algorithme :

Initialisation initialisation de k centres $(cl_1, ..cl_k)$ dans X.
Mise à jour tant que $\Delta(cl_1, ...cl_k) > 0$, assigner à chaque centre les points de sa cellule de Voronoi. Mettre à jour la position de chaque centre comme le barycentre de sa cellule de Voronoi courante.

La complexité de l'initialisation est en $O(k * d)$: la position de chacun des k centres, de dimension d, est choisie aléatoirement. à chaque itération, la constitution des cellules de Voronoi est l'étape plus coûteuse : pour chaque vecteur $x \in X$, on cherche le centre voisin le plus proche en $O(d * n * k)$. La complexité de l'algorithme des k-moyennes sous sa forme la plus élémentaire est donc en $O(d * n * k)$ par itération.

Du point de vue théorique, il est difficile d'anticiper le nombre d'itérations nécessaires pour atteindre une position d'équilibre. Toutefois, Bengio et Bottou montrent que l'heuristique des k-moyennes consiste exactement à suivre le gradient de la RSS 3.1 (Bottou et Bengio, 1995). Par conséquent, les centres convergent rapidement vers une position d'équi-

libre. Toutefois, cet équilibre local reflète rarement la réalité de la configuration optimale globale. Aussi des approches ont-elles été introduites pour pâlier au problème des minimums locaux de l'algorithme des k-moyennes.

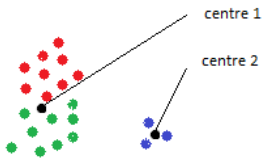

Figure 3.3 Exemple de minimum local dans le cadre de l'algorithme des k-moyennes

3.1.3 Méthodes déterministes versus approches randomisées

Face à l'insuffisance de l'heuristique des k-moyennes, des approches aussi diverses que les heuristiques randomisées, les algorithmes déterministes approchés, ou encore les réseaux de neurones ont été proposés pour réaliser le clustering d'un ensemble de données.

Approches déterministes. En 2000, J. Matousek introduit une procédure de discrétisation de l'espace permettant de rechercher la position optimale des centres dans un ensemble de cardinal fini (Matousek, 2000). Les positions candidates sont obtenues en subdivisant l'espace de manière à construire des k-tuples de centres bien repartis (ϵ-répartition) pour assurer une limitation de l'erreur en ϵ. Une telle subdivision suivie d'une recherche exhaustive garantit une $(1+\epsilon)$-approximation du schéma de clustering optimal. Malheureusement, cet ensemble croît en exponentiel de k, le nombre de centres. La méthode est donc inapplicable pour k élevé. Comme le souligne Matousek lui-même, son étude résulte de motivations théoriques plus que pratiques (la complexité finale est en $n * log(n)^k * \epsilon^{-2k^2 d}$, avec n le cardinal de l'espace et d la dimension). Plus tard, Kanungo et al. parviennent à exploiter l'idée de Matousek pour dresser une liste raisonnable de centres candidats à l'optimum, au prix d'une perte de précision (Kanungo *et al.*, 2004). Ils montrent empiriquement que leur méthode conduit à une amélioration des performances de l'algorithme des k-moyennes élémentaire.

Heuristiques randomisées. Les heuristiques randomisées ne procurent pas de garantie quant à la précision du résultat. Elles permettent néanmoins d'obtenir en un temps raisonnable une configuration acceptable. En terme d'heuristique randomisée à opérateur de recherche locale appliqué au problème des k-moyennes, on cite souvent la tentative recuit simulé de Shokri et al., qui obtiennent des résultats encourageants sur plusieurs jeux de données

(Selim et Alsultan, 1991). Toutefois, la procédure de discrétisation de l'espace est bien trop complexe pour les problèmes de grande dimension. L'auteur propose en effet de discrétiser chaque dimension en s segments pour construire un ensemble discret de centres candidats. Ce procédé impose soit une perte de précision conséquente, soit un problème de complexité considérable. L'approche évolutionniste de Painho et Bacao est également limitée aux petits problèmes (Painho et Bacao, 2000). De manière générale, pour des espaces de dimension élevée, l'espace de recherche des positions des centres est si vaste que les heuristiques à opérateur de recherche locale s'essoufflent rapidement. En pratique, on renonce donc au calcul explicite d'un voisinage. L'algorithme des k-moyennes de Lloyd correspondant à une descente pure, on se contente souvent de la répéter (descente avec relance) pour éviter les minimums locaux.

Ici, nous proposons une méthode d'identification de configurations critiques, permettant d'améliorer la procédure de descente de l'algorithme des k-moyennes. En termes d'efficacité, notre méthode dépasse signicativement la méthode de Kanungo et al., sans porter préjudice à la qualité du clustering.

3.1.4 Facteurs critiques

Initialisation des centres. Dans le cadre de l'algorithme des k-moyennes, l'initialisation de la position des centres ne peut être réalisée aléatoirement sous peine de générer des points situés très loin de la distribution des données. On choisit donc en général d'initialiser les centres en utilisant des points existants, choisis aléatoirement dans X. Toutefois, malgré cette précaution, rien ne garantit que les centres soient régulièrement distribués sur la distribution à estimer. Arthur et Vassilvitski proposent une procédure d'initialisation plus fine, qui permet de pâlier à cet inconvénient(Arthur et Vassilvitskii, 2006). Si $D(x)$ est la plus petite distance de x a l'un des centres deja choisis, on procède comme suit :

choisir $cl_1 = x$ aléatoirement dans X.

pour i $\in [1, k]$

assignation : choisir $cl_i = x'$, x' choisi aléatoirement dans X-x avec la probabilité $\frac{D(x')^2}{\sum_{x \in X-x} D(x)^2}$

On obtient alors une garantie sur le résultat (Arthur et Vassilvitskii, 2006) en $O(log(k))$ du clustering optimal. Sous cette forme, l'algorithme est communément appelé kmeans++.

Poids des clusters. Le poids relatif des clusters constitue un autre facteur critique que le bon sens suggère de contrôler (bien que ceci puisse pénaliser l'optimum dans des configurations extrêmes). Une configuration problématique patente dans le cadre des k-moyennes consiste en effet en un centre accaparé par quelques points situés loin du reste de la distribution. Une heuristique raisonnable consiste donc à encadrer le poids des clusters, pour garantir une bonne répartition des centres sur la distribution des données. Si la RSS (équation 3.1)

s'en trouve éventuellement pénalisée, dans le contexte de la constitution d'un vocabulaire, cette exception est peu gênante puisqu'on ne souhaite pas prendre en compte les descripteurs SIFTs isolés.

3.1.5 Notre implémentation

Volet non hiérarchique

Initialisation. L'initialisation est réalisée selon la stratégie proposée par Arthur et Vassilvitskii Dans la mesure où nous choisissons $k << d$, la pression sélective imposée sur les centres reste faible et l'accroîssement de la complexité du à ce changement est négligeable. L'initialisation des centres reste en $O(k * d)$.

Mise à jour des positions des centres. Comme dans l'algorithme des k-moyennes classiques, à chaque itération, les centres se voient associer les points contenus dans leur cellule de Voronoi. Suite à la mise à jour de la position des centres, on calcule alors le mouvement cumulé ($\Delta_m = \sum_j \left\| cl_j^{previous} - cl_j^{new} \right\|$), qui sert de critère d'arrêt.

Rééquilibrage. Nous avons déjà évoqué le cas critique d'un cluster très compact constitué de quelques centres. Trivialement, la présence de petits clusters implique que les clusters restants sont relativement lourds. Les petits clusters correspondent à des détails de la distribution, rendu trop précisément au dépend de la réalité globale.

Une stratégie naïve pour prévenir cette configuration consiste à imposer une borne inférieure au poids des clusters :

$$\forall j,\, w(Cl_j) < s \tag{3.6}$$

Si les données sont régulièrement réparties dans l'espace, ce type de limitation est susceptible d'améliorer les résultats obtenus par l'algorithme de Lloyd élémentaire. Toutefois, la distribution des descripteurs SIFTs extraits sur des images est rarement régulière. Au contraire, on observe notamment des clusters très denses, constitués des descripteurs très fréquents (faible courbure, coins, textures récurrentes). Attribuer trop de centres à de tels descripteurs revient à attribuer deux mots visuels différents à des éléments peu informatifs, car trop fréquents dans la distribution. La discrimination par le poids n'est donc pas une heuristique satisfaisante.

On défini donc la RSS par élément :

$$r(Cl_j) = \frac{f_{RSS}(Cl_j)}{w(Cl_j)}, \tag{3.7}$$

avec $w(Cl_j)$ le poids (nombre d'éléments) du cluster Cl_j,

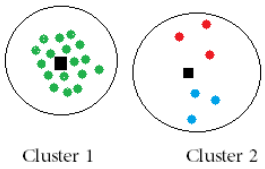

Figure 3.4 Limites de la discrimination par le poids des clusters

A l'inverse du poids, la RSS par élément permet de distinguer les clusters selon leur compacité. Aussi, proposons nous de détecter les clusters critiques vérifiant :

$$r(Cl_i) > \tau * min_j(r(Cl_j)),\ j \in J,\ w(Cl_j) > s \tag{3.8}$$

avec $s << n$. Le centre i est alors réinjecté dans le cluster j' avec la probabilité $P(r(Cl_{j'}))$:

$$P(r(Cl_{j'})) = \frac{(r(Cl_{j'}) - \tau_p)^2}{(r(Cl_{jmin}) - r(Cl_{jmax}))^2} \tag{3.9}$$

On prendra par exemple : $\tau_p = 0.5 * (r(Cl_{jmin}) - r(Cl_{jmax}))$.

Volet hiérarchique

Jusqu'à présent nous n'avons pas discuté de la complexité de notre implémentation et de son aptitude à s'adapter aux grands jeux de données. Le goulot d'étranglement dans la mise en oeuvre de l'algorithme des k-moyennes correspond à l'assignation de chacun des vecteurs x de X à son centre le plus proche ($O(n * k * d)$).

Récemment, de nombreux articles (Kanungo *et al.* (2004)), ont fait référence à l'utilisation d'un kd-arbre pour le calcul du contenu des cellules de Voronoi de chaque centre. Ainsi, l'algorithme de Kanungo et al. repose sur la construction d'un kd-arbre sur les points de X. à chaque itération, la recherche du plus proche voisin de chaque point est réalisée en propageant dans l'arbre une liste de centres candidats. Cet algorithme, dit de filtrage, permet d'accélérer sensiblement le calcul de l'étape la plus lourde de l'algorithme des k-moyennes.

Quitte à utiliser une structure arborescente, nous préférons utiliser, à l'instar de Mikolajczyk et Nister (Mikolajczyk (2006), Nister et Stewenius (2006)), une approche hiérarchique de clustering. La procédure consiste à construire un p-arbre en procédant à une division récursive des noeuds via l'algorithme des k-moyennes (avec $p = k$).

Quelle que soit l'heuristique utilisée pour la scission des noeuds, du fait de l'imperfection

inéluctable du partitionnement de l'espace à chaque noeud, le recours à une structure hiérarchique entraîne des imprécisions importantes lors de la recherche du plus proche voisin (voire paragraphe 2.3).

Deux stratégies élémentaires permettent de remédier à cette situation : l'utilisation d'un facteur de partitionnement plus élevé ou le recours à la notion d'appartenance floue pour les points ambigus (approche fuzzy c-means, ou dans une optique plus structurelle, *spill trees*). Ici, nous mettons en place une procédure de partitionnement à p variable afin de minimiser l'erreur commise lors des premières partitions, sans pénaliser la complexité outre mesure. Nous montrons, comme le soulignent Schindler et al., qu'en utilisant un facteur de partitionnement suffisamment grand au départ, on peut réduire considérablement l'imprécision commise (G. Schindler, 2007).

Comme le soulignent Philbin et al., le processus hiérarchique permet d'accélérer fortement la procédure horizontale (Philbin *et al.*, 2007). La construction d'un arbre équilibre de profondeur l, avec un facteur de partitionnement p, est en :

$$O(\sum_{i \in 1..l} \frac{n}{p^i} * d * p * p^i) \tag{3.10}$$

Or, si l'arbre est équilibré, k étant le nombre de feuilles obtenues au terme du processus :

$$l = log(\frac{k}{p}). \tag{3.11}$$

D'où :

$$\sum_{i \in 1..l} n * d * p = log(\frac{k}{p}) * n * d * p. \tag{3.12}$$

Finalement, la complexité de la procédure est en :

$$O(n * d * p * log(\frac{k}{p})) << O(n * k * d) \tag{3.13}$$

3.1.6 Résultats expérimentaux et discussion

Résultats expérimentaux

Nous avons déjà mentionné à plusieurs reprises la librairie de Kanungo et al. (Kanungo *et al.*, 2004). [1]Elle constitue notre base de comparaison expérimentale. Ci-après, nous présentons les résultats obtenus. La qualité du clustering est mesurée au moyen de la RSS (équation 3.7), qui mesure l'erreur commise par élément.

1. http ://www.cs.umd.edu/mount/Projects/KMeans/

Tableau 3.1 Construction d'un vocabulaire visuel : procédure de recherche locale de Kanungo et al. versus notre implémentation (RSSBFLAT), appliquée à GoogleDB (184k descr., 128 dim)

Base de donnée	methode	numClusters	NbIterMax	RSS par élément ($.10^4$)	t(s)
GoogleDB	RSSBFLAT	100	300	7.49	108
GoogleDB	Kanungo *et al.*	100	300	7.43	4122
GoogleDB	RSSBFLAT	500	300	6.36	123
GoogleDB	Kanungo *et al.*	500	300	6.11	7187
GoogleDB	RSSBFLAT	1000	300	2.98	582
GoogleDB	Kanungo *et al.*	1000	300	5.62	18650
GoogleDB	RSSBFLAT	2500	300	2.79	979

Tableau 3.2 Construction d'un vocabulaire visuel : YoutubeDB (889k*128)

Base de donnée	methode	numClusters	NbIterMax	RSS par élément ($*10^4$)	t(s)
YoutubeDB	RSSBFLAT	100	300	8.78	904
YoutubeDB	Kanungo *et al.*	100	300	8.65	11200
YoutubeDB	RSSBFLAT	500	300	1.58	1027
YoutubeDB	Kanungo *et al.*	500	300	7.56	44750
YoutubeDB	RSSBFLAT	1000	300	0.8	2567
YoutubeDB	Kanungo *et al.*	1000	300	-	$> 8h$

Tableau 3.3 Construction d'un vocabulaire visuel : PascalDB (1.9M*128)

Base de donnée	methode	numClusters	NbIterMax	RSS par élément ($.10^4$)	t(s)
PascalDB	RSSBFLAT	100	300	18.2	7124
PascalDB	RSSBFLAT	500	300	15.8	8318
PascalDB	RSSBFLAT	1000	300	6.46	9456
PascalDB	RSSBFLAT	5000	300	0.42	9242

Sur notre plus petite base de données, GoogleDB (Tableau 3.1), l'algorithme de Kanungo et al. obtient une RSS légèrement inférieure, pour un temps d'exécution largement supérieur. Cette tendance se vérifie sur la base de données YouTubeDB (Tableau 3.2) pour 100 centres. Cependant, à 500 centres, notre algorithme dépasse celui de Kanungo et al. en précision. Enfin, sur la base de données PascalDB (Tableau 3.3), le temps d'exécution de l'algorithme de Kanungo et al. est trop élevé pour être utilisé en pratique, tandis que notre algorithme procure un résultat raisonnable en moins de trois heures.

Nous abordons à présent l'évaluation de notre procédure hiérarchique. Au paragraphe précédent (3.1.5), nous avons souligné l'avantage d'un arbre à facteur de partition p variable. Le tableau 3.4 compare la RSS obtenue avec un facteur de partitionnement (p) fixe à la RSS

obtenue avec un p variable décroissant. Nous montrons empiriquement que l'utilisation d'un p dynamique décroissant permet d'améliorer la précision du schéma de clustering obtenu.

Tableau 3.4 Clustering hiérarchique, influence d'un facteur de partitionnement dynamique, GoogleDB (884k*128)

Base de donnée	p	profondeur (l)	p^l	RSS $*(10^4)$	t(s)
GoogleDB	50	2	2500	0.69	1602
GoogleDB	100-25	2	2500	0.56	6810
GoogleDB	20	3	8000	0.38	1061
GoogleDB	100-80	3	8000	0.22	2658

Nous appliquons à présent, tableau 3.5, la stratégie de clustering hiérarchique aux bases de données YoutubeDB et PascalDB. Il est intéressant de noter que la position du point d'inflexion de la courbe RSS par élément-nombre de clusters semble se situer entre 1000 et 10,000 descripteurs pour PascalDB comme pour GoogleDB, bien que le ratio $n/|c|$ (n le nombre de descripteurs, $|c|$ nombre de classes), soit fondamentalement différent.

Tableau 3.5 Clustering hiérarchique, vocabulaires hiérarchiques, GoogleDB (884k*128)

Base de donnée	p	l	nbDescr	% N_d	RSS $*(10^4)$	t(s)
GoogleDB	100-100	2	10000	7	-	4197
GoogleDB	100-75	2	7500	5	0.23	4011
GoogleDB	100-50	2	5000	2.5	0.56	2860
GoogleDB	100-25	2	2500	1	0.71	2571
PascalDB	100-50	2	5000	0.25	0.33	53589
PascalDB	200-50	2	10000	0.5	0.26	71782
PascalDB	200-100	2	20000	1	0.17	75535
PascalDB	200-200	2	40000	2	-	60958
PascalDB	100-50-20	3	100000	5	-	34987

Discussion et conclusion partielle

La construction d'un vocabulaire visuel est une étape clé de la procédure d'apprentissage en traitement d'images. Elle constitue l'un des goulots d'étranglement du processus lors du traitement de bases de données à grande échelle. De plus, les choix réalisés à ce stade en terme de dimensionnalité impactent les étapes suivantes, lors du choix d'une stratégie d'encodage et d'un algorithme de classification. En effet, opter pour un vocabulaire de très grande taille limite la complexité des algorithmes utilisés par la suite. Aussi, est-il légitime de s'interroger quant à la taille appropriée d'un vocabulaire visuel.

Figure 3.5 RSS par élément (gauche : GoogleDB, droite : PascalDB)

Obj1.1.i : Mettre en évidence un critère théorique permettant de justifier le choix de l'échelle des vocabulaires visuels.

En ce qui concerne la taille du vocabulaire, nous avons montré empiriquement que la diminution de la RSS par élément (équation 3.7) n'est pas linéaire en fonction de la taille du vocabulaire. Il semble plutôt qu'à partir d'un certain point, l'ajout de nouveaux centres n'améliore pas beaucoup la description de la distribution. Pour choisir la taille du vocabulaire visuel, on pourra opter pour le point d'inflexion de la courbe de la RSS par élément.

Obj1.1.ii : Proposer des solutions pour la construction de vocabulaires visuels sur des bases de données à grande échelle.

Dans ce paragraphe, nous avons implémenté une méthode capable de réaliser un k-clustering non supervisé dépassant les performances de la librairie de Kanungo et al. La complexité de notre méthode (RSSBFLAT) est en $O(n * k * d)$. En pratique, cette méthode est applicable au problème de k-clustering pour $k << n$. Nous avons également proposé des solutions pour le problème de k-clustering pour des k plus élevés, reposant sur une procédure hiérarchique. La complexité de la procédure est alors en $O(n * d * p * log(\frac{k}{p})) << O(n * k * d)$ pour $k >> 100$. Nous avons alors souligné l'avantage que l'on peut tirer de l'utilisation d'un facteur de partitionnement dynamique décroissant dans le cadre du clustering hiérarchique.

3.2 Encodage des images

3.2.1 Panorama des méthodes d'encodage des images au moyen d'un vocabulaire visuel

L'encodage d'une image sous forme d'un histogramme de mots visuels (les centres définis plus haut) permet de réduire la dimension de la représentation d'une image, modulo une importante perte d'information. L'étude menée en 3.1 souligne que l'accroissement de la taille du vocabulaire visuel ne permet pas de réduire la perte de précision due à l'encodage en

histogrammes de mots visuels. Dans ce paragraphe, nous passons en revue diverses stratégies d'encodage des images au moyen d'un vocabulaire visuel. Nous examinons leur portabilité vers la grande échelle et leur impact sur la fidélité de la représentation des images.

L'encodage binaire constitue le standard en terme d'encodage d'image sous forme de sacs de mots visuels. Il s'agit simplement d'associer chaque descripteur à son plus proche voisin dans le vocabulaire visuel. Un histogramme de la taille du vocabulaire est constitué en comptant pour chaque mot visuel le nombre de descripteurs voisins contenus dans une image. Cette procédure peut être réalisée en $O(n * m * d)$, avec $n = \|X\|$ le nombre de descripteurs, m la taille du vocabulaire visuel, et d la dimension des vecteurs de X. Cette représentation souffre de plusieurs manquements (Figure 3.6), ce qui a conduit les chercheurs à élaborer des méthodes plus fines d'encodage.

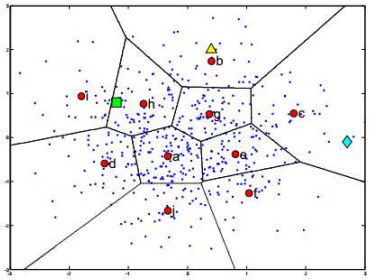

Figure 3.6 Ambiguité de l'approche d'encodage binaire par mots visuels [41]. Les points rouges représentent les mots visuels issus de la procédure de clustering. Le triangle jaune illustre un descripteur bien encodé par le vocabulaire. Le carré vert et le losange bleu représentent deux situations critiques où les descripteurs sont mal représentés.

Gemert, Veenman, Smeulders et Geusebroek comparent plusieurs stratégies d'encodage des descripteurs basées sur la notion de *soft-encoding* (Figure 3.7, (van Gemert *et al.*, 2010)). Ils mettent notamment en évidence le cas critique des descripteurs peu probables (descripteurs loins de tous les mots du vocabulaire visuel), et le cas critique des descripteurs ambigus (descripteur à mi-chemin entre deux mots du vocabulaire visuel). Afin de remédier à ces deux écueils de la représentation binaire, ils proposent trois méthodes basées sur l'encodage des descripteurs SIFTs au moyen de plusieurs mots visuels, en prenant en compte la distance respective du descripteur à chacun des éléments du vocabulaire.

Dans la même optique, l'approche des mots visuels flous(Bouachir *et al.* (2009)) est basée sur la fonction d'appartenance introduite dans le cadre des FuzzyC-Means par Bezdeck (Bezdek, 1981). Plutôt que de représenter un descripteur via une association binaire à un mot du corpus, on le modélise par son degré d'appartenance à ses cellules de Voronoi voisines :

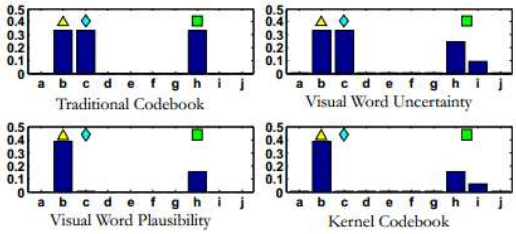

Figure 3.7 Stratégies d'encodage (van Gemert *et al.*, 2010). De gauche à droite et de haut en bas l'encodage binaire, l'encodage de l'incertitude (*codebook uncertainety*), l'encodage de la plausibilité (*codebook plausibility*), et l'encodage par noyau (*kernel codebook*)

$$h_{ij} = \frac{1}{\sum_{m \in 1,k} (\frac{\|x_i - v_j\|}{\|x_i - v_m\|})^{\frac{2}{f-1}}} \tag{3.14}$$

avec h_{ij} la composante associée au mot visuel j pour le descripteur x_i, $(v_1, .., v_k)$ étant les k centres les plus proches de de x_i dans le corpus visuel.

Dans une optique plus formelle, des stratégies d'optimisation ont été mise en oeuvre pour améliorer l'encodage des descripteurs SIFTs au moyen d'un vocabulaire visuel. Pour comprendre ces approches, il s'agit de remarquer que la stratégie d'encodage binaire des descripteurs via un vocabulaire visuel peut être formulée comme un problème d'optimisation sous contraintes. Ainsi, si on note c_i l'histogramme associé au descripteur x_i, B la matrice dont les colonnes représentent les mots du vocabulaire, optimiser c_i revient à estimer [2] :

$$\min_c \sum \|x_i - Bc_i\|^2 \text{ s.c. } \|c_i\|_0 = 1 et \|c_i\|_1 = 1. \tag{3.15}$$

Dans le cadre du LLC (Locally constrained Linear Coding (Wang *et al.*, 2010)), le problème est alors relaxé en :

$$\min_c \sum \|x_i - Bc_i\|^2 + \lambda \|d_i \odot c_i\|^2 \text{ s.c. } 1'c_i = 1, \text{ avec } d_i = exp(\frac{dist(x_i, B)}{\sigma}) \tag{3.16}$$

Cette formulation assouplit les contraintes sur l'assignation des descripteurs aux mots du corpus, en autorisant l'utilisation de plusieurs mots pour maximiser la précision de l'encodage (composante $\|x_i - Bc_i\|^2$). La distance du descripteur à chaque mot utilisé agit, elle, comme

2. pour faciliter la lecture, 1 est ici tantôt l'identité vectorielle tantot l'identité matricielle

une pénalité sur la fonction objectif (composante $\lambda \|d_i \odot c_i\|^2$). Ainsi formulé, le problème peut être résolu de façon analytique :

$$(C_i + \lambda * diag(d)) * v_i = 1 \tag{3.17}$$

Avec :

$$c_i = \frac{v_i}{\|v_i\|}, \text{ et } C_i = (B_k - 1X_i')(B_k - 1X_i')', \ d_i = exp(\frac{dist(x_i, B)}{\sigma}) \tag{3.18}$$

La complexité de la procédure est alors en $O(m^{i\in[2,3]})$ pour chaque descripteur, du fait de l'inversion de $(C_i + \lambda * diag(d))$, avec m est la taille du vocabulaire visuel ($i = 3$ avec Gauss-Jordan, et de manière générale $i > 2$). Cette valeur est trop importante pour un usage du procédé à grande échelle. Cependant, comme le suggèrent Wang et al., on peut appliquer une version allégée du LLC en ne considérant que les k mots voisins (Wang *et al.*, 2010) :

$$\min_C \sum \|x_i - c_i * B_k\|^2 \text{ avec, } 1'c_i = 1 \tag{3.19}$$

B_k est la matrice de k colonnes constituée des k voisins de x_i. La complexité est alors en $O(n * k^2)$ par descripteur, avec $k << m$.

L'alternative a ce type d'encodage compact consiste à utiliser une représentation appelée vecteur de Fisher, introduite par Perronnin *et al.* (2010). Pour un vecteur de Fisher, la dimension de la représentation des images est de l'ordre de plusieurs centaines de milliers de dimensions. Dans ce dernier cas de figure, il est nécessaire d'avoir recours à des techniques avancées de compression et d'indexation (LSH et variantes).

3.2.2 Encodage d'images avec arrière-plan

Position du problème

Si l'on souhaite obtenir un nombre conséquent d'images d'entraînement de manière automatisée, il est nécessaire de tenir compte du fait que les images sont susceptibles de présenter l'objet dans un contexte parasitaire. Or, la présence d'arrière-plan nuit à l'apprentissage du fait du phénomène d'achalandage d'histogramme (*background clutter*). S'il est possible d'obtenir des images sans arrière-plan (PNGs) en réalisant un filtrage sur Google, celles-ci viennent en nombre limité. De plus, la robustesse d'une procédure d'apprentissage dépend fortement de la ressemblance des images d'entraînement aux images de test. En particulier, la base de donnée d'apprentissage doit être représentative de la diversité intraclasse des objets d'intérêt. Par exemple, les exemplaires de la base de données dédiée à la reconnaissance de

guitares doit contenir des manches de guitare, des corps de guitare, et présenter les guitares aux mains d'un musicien, les doigts occludant les cordes.

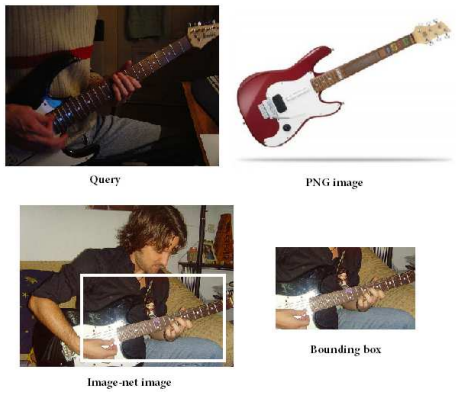

Figure 3.8 Choisir une bonne base de données d'apprentissage (haut-gauche : image inconnue, haut-droite : PNG, bas-gauche : image issue d'image-net, bas-droite : bon exemplaire pour l'apprentissage)

Les techniques d'encodage décrites en 3.2.1 sont toutes sensibles au phénomène d'achalandage d'histogramme. L'apprentissage d'un classificateur sur des images de type image-net encodées sous forme de mots visuel est pénalisé par la présence d'un ensemble parfois conséquent de descripteurs n'appartenant pas à l'objet d'intérêt. De même, au moment du test, si l'objet occupe un espace très réduit sur l'image, il est peu probable qu'il soit détecté. Par ailleurs, lorsque la frontière entre deux classes est très fine (ex : trompette, saxophone), la différence entre les deux classes est susceptible de devenir négligeable par rapport aux phénomènes parasitaires dus à l'arrière-plan.

En détection d'objets (i.e lorsque la position de l'objet doit être détectée), les images d'entraînement sont assorties d'une boîte englobant l'objet. Malheureusement, l'estimation manuelle des boîtes englobantes est un processus coûteux, non adapté aux bases de données à grande échelle. Dans la suite de ce paragraphe, nous nous attachons à concevoir une méthode automatisée d'estimation de boîtes englobantes, permettant de séparer sur les images d'entraînement les zones recouvrant l'objet et les zones contenant de l'arrière-plan.

Analyse d'images par région

La première intuition lorsqu'on recherche à résoudre le problème d'achalandage d'histogramme consiste à se tourner vers les méthodes de segmentation. Récemment, des résultats très encourageants ont été obtenus dans ce domaine. Notamment, Bach et Ponce marient une

approche de clustering discriminant (classification) et une approche de coupure normalisée (segmentation), pour segmenter des images à partir d'un étiquetage faible (sans boîte englobante) (Bach et Ponce, 2010). Leur méthode tire partie des points communs des images d'une même classe pour guider la procédure traditionnelle de coupure normalisée (*normalized cut*). On parle de cosegmentation. Toutefois, le travail s'effectue au niveau des super pixels et la complexité de la procédure est en $O(s^2)$, s étant le nombre de superpixels. En 2012, Xing et Kim (Kim et Xing, 2012) soulignent que la segmentation donne lieu à l'identification de régions connexes, ce qui n'est pas approprié dans le cadre d'une procédure d'apprentissage en classification.

Les techniques de détection d'objet ou de classification fine sont plus à même de répondre au problème qui nous intéresse ici. Ainsi, notre procédure se rapproche plus des travaux de Kim et Torralba en 2006 (Kim et Torralba, 2006), de Yao et al. en 2011 (Yao *et al.*, 2011a) ou de Jia, Huan et Darrell en 2012 (Jia *et al.*, 2012). L'idée directrice de ces travaux consiste à apprendre les zones pertinentes pour la classification dans une image. Jia, Huan et Darrell obtiennent l'état de l'art sur la base de données CIFAR-10 en 2012. Ils expliquent que la technique d'encodage standard n'est pas adaptée aux images naturelles (*natural images*), c'est-à-dire les images avec une quantité considérable d'arrière-plans. Ils proposent alors d'apprendre conjointement les régions pertinentes d'une image et un classificateur linéaire d'objets. Ils soulignent également que cette analyse fine permet d'utiliser un vocabulaire plus restreint sans dégrader les performances. Plus précisément, leur procédure consiste à segmenter l'image en un ensemble de régions $R = (R_1, .., R_M)$ (ensembles de pixels) et à maximiser la probabilité du jeu de données d'entrainement (X, Y) (X les images, Y les étiquettes associées) obtenue via un classificateur linéaire entraîné sur ces régions :

$$min_{c,R,\theta} \frac{1}{N} \sum_{n=1:N} L(f(x_n, \theta), y_n) + \lambda Reg(\theta) \tag{3.20}$$

avec $x_{ni} = op(A^{ci}_{n,R_i})$, $y = f(x, \theta)$, A^{ci}_{n,R_i} l'activation du mot visuel c_i dans la region R_i, op un opérateur de lissage (moyenne) sur la région pour l'image n. La formulation du problème est élégante, cependant, sous ses abords séduisants (convexité), la fonction objectif (équation 3.20) est impraticable à grande échelle.

La démarche proposée par Torralba et al. est moins laborieuse. Elle consiste à raffiner itérativement la définition de la région d'intérêt pour chaque image. Ainsi, au début du processus, toutes les images ont un score égal et constitue H, l'ensemble d'images de référence. Ensuite, à chaque itération i, les régions présentant le plus de ressemblance avec H_i sont sélectionnées et constituent le nouvel ensemble de référence H_{i+1}. Au terme du processus, des régions pertinentes pour la classification sont identifiées sur chaque image. Les résultats

obtenus sont de l'ordre de l'état de l'art en 2006 en détection sur trois objets de la base de données.

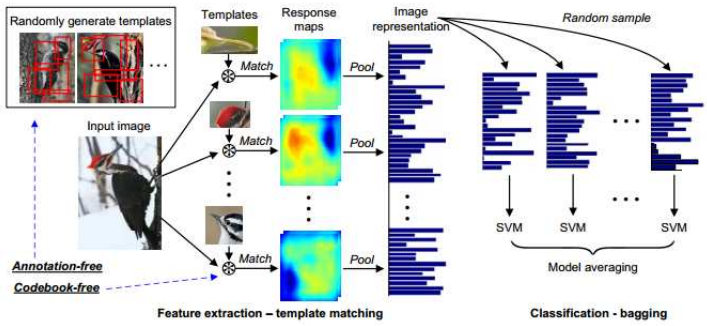

Figure 3.9 Sélection de zones discriminantes dans une image (Yao *et al.*, 2011a)

Enfin, la démarche de Yao et al. en 2011 consiste à identifier les zones d'intérêt dans une image en générant un ensemble de régions d'intérêt et en notant chaque région par sa similarité avec les autres régions de la même classe d'image. La procédure présentée dans (Yao *et al.*, 2011a) est optimisée pour la reconnaissance d'images présentant des différences subtiles (deux races d'oiseaux). La classification est réalisée au moyen de SVMs linéaires, combinés selon une fonction objectif visant à en minimiser la corrélation. La démarche est motivée par le fait qu'un seul SVM est peu susceptible d'apprendre une frontière suffisamment précise entre des classes très semblables, notamment en présence de régions fortes relativement différentes pour chaque classe (pour les oiseaux, le bec, les pattes).

Dans un autre article publié en 2011, Yao et al proposent de réaliser la classification des régions informatives au moyen d'une forêt randomisée (Yao *et al.*, 2011b) d'arbres de décision discriminatifs. Un arbre de décision discriminatif diffère d'un arbre de décision classique par le mode de la scission des noeuds. Dans un arbre classique, les noeuds sont séparés selon une heuristique (médiane, information gain) faible. Dans un arbre de décision discriminatif, les noeuds sont séparés au moyen d'un classificateur fort (pour nous et Yao. et al., une SVM). Nous reviendrons plus en détail sur cet article en section 3.3 concernant la classification.

3.2.3 Strategie proposée

L'objet de la procédure décrite ici est d'identifier sur une image d'entraînement annotée avec la classe $c_j \in C = (c_1, ..., c_{N_c})$ (annotation faible) quelles sont les zones informatives pour la classification (zones contenant l'objet) et quelles sont les zones qui peuvent être négligées.

En d'autres termes, nous essayons d'estimer automatiquement la boîte englobante d'un objet.

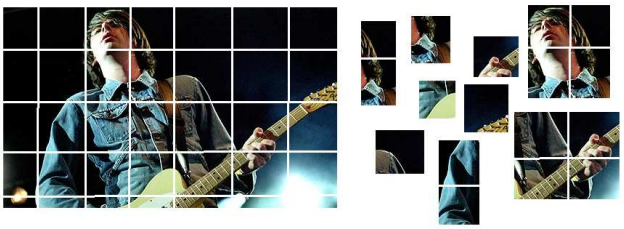

Figure 3.10 Subdivision de l'image

Subdivision de l'image. L'image est découpée selon plusieurs grilles régulières de pas décroissant. Pour chaque cellule de ces grilles (ou région), on construit un histogramme de mots visuels selon la procédure d'encodage LLC décrite plus haut (équation 3.19). Certaines de ces régions contiennent l'objet ou une partie de l'objet d'intérêt, certaines ne contiennent pas l'objet. Par la suite, nous notons $H_i = (h_{i1}, ...h_{iR_i})$ l'ensemble des R_i histogrammes associés aux régions de l'image i. En pratique, la construction des histogrammes est optimisée en calculant d'abord les histogrammes associés aux régions de petite taille, dont les contributions sont additionnées pour former les histogrammes des régions plus vastes.

Identification des zones pertinentes d'une image. Notre stratégie pour la sélection de régions pertinentes consiste à procéder au clustering hiérarchique des régions extraites sur les images d'entrainement $H = \cup_{i \in (h_{i1}, ...h_{iR_i})} H_i$. La motivation de cette démarche est de faire en sorte que les régions visuellement similaires se regroupent en amas. Les régions informatives pour une classe d'objets se trouvent alors dans des clusters purs (clusters composés en majorité d'une seule classe d'objet), tandis que les régions non signifiantes sont situées dans des clusters disparates.

Nous formalisons à présent l'intuition exposée plus haut. On note $w_1, ...w_k$ les clusters formés sur H. N_H est le nombre total de régions, N_c le nombre de classes, $|c_i|$ le cardinal de la classe c_i et $|w_i|$ le cardinal du cluster i. Comment évaluer la pertinence d'une région d'une image donnée étant donné un arbre de clusters ?

Nous proposons ici d'utiliser la notion de gain d'information mutuelle. Expliquons pourquoi. étant donné un arbre de régions étiquetées, il s'agit d'attribuer à chaque région un indicateur de pertinence. Pour ce faire, nous nous proposons comme Yao et al d'utiliser la ressemblance de cette région aux images de la même classe. En terme de clustering, cela signifie que la région appartient à un cluster composé de nombreuses régions de sa classe (les

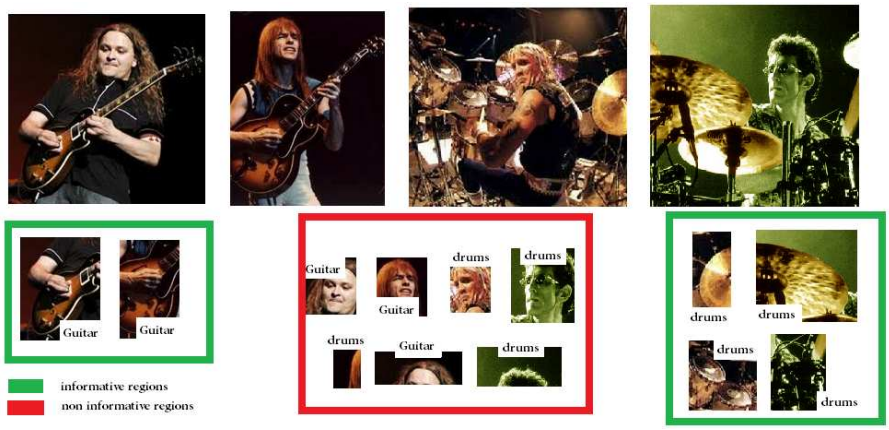

Figure 3.11 Sélection de zones discriminantes

manches de guitare, les clés d'accord, les cymbales). On peu donc se référer à la pureté de la feuille finale dans l'arbre des clusters :

$$pur_{w_j c_i} = \frac{\|w_j \cap c_i\|}{\|w_j\|} \tag{3.21}$$

Toutefois, cet indicateur est insuffisant, car, si l'on pousse la procédure de scission de l'espace à l'extrême, on obtient des feuilles éventuellement unitaires, dont la pureté n'est pas signifiante. Il s'agit donc de prendre également en compte la représentativité d'un cluster vis-à-vis d'une classe donnée.

$$r_{wc} = \frac{\|w \cap c\|}{\|c\|} \tag{3.22}$$

Le critère d'information mutuelle (équation 3.23) traduit l'intuition que l'on a de recourir à un compromis entre ces deux mesures :

$$\mathsf{IG}(w,c) = P(c,w) log\left(\frac{P(c,w)}{P(c)P(w)}\right) = \frac{\|w \cap c\|}{N} log\left(\frac{N\|w \cap c\|}{\|w\|\|c\|}\right), \tag{3.23}$$

où :

$$\frac{\|w \cap c\|}{N} \text{ mesure la représentativité,} \tag{3.24}$$

et

$$\frac{w \cap c}{w} \text{ mesure la pureté.} \tag{3.25}$$

Enfin,

$$\frac{c}{N} \text{ représente la probabilité de la classe c.} \tag{3.26}$$

Une région h_i se voit alors attribuer le score maximal parmi les clusters situés sur sa trajectoire T_{hi} de la racine de l'arbre de clusters vers la feuille que la région occupe au terme de la construction de l'arbre. Ainsi, on calculera dynamiquement le score $sc(h_i, c)$ (IG) associé à la région h_i de la classe c comme :

$$sc(h_i, c) = max_w(IG(w, c)),\ w \in T_{hi}. \tag{3.27}$$

Pour chaque image de la base de données d'entraînement, il est alors possible d'estimer automatiquement une fenêtre englobant l'objet. Par exemple, on choisira le rectangle englobant les régions de score maximal de chaque image (équation 3.27).

Figure 3.12 Exemple de fenêtre englobantes construites sur des images d'image-net.

Nous avons ainsi distingué les régions de chaque image qui contiennent l'objet d'intérêt. En apprenant une SVM par classe associée à cette fenêtre englobante, on observe une amélioration de 2% du F1-score par rapport à une stratégie de classification identique sans estimation de la fenêtre englobante. Ce résultat est positif, mais pas suffisant pour clamer une amélioration significative. Le fait est que nous essayons de classer toutes les images (ou régions d'images) de guitare au moyen d'un seul hyperplan. La diversité de ces régions (caractère multinomial) est susceptible de nuire à l'apprentissage. Nous présentons au chapitre suivant un classificateur plus robuste permettant de tirer un meilleur parti des régions discriminantes

identifiées et de mieux supporter la multinomialité des classes d'objets.

3.2.4 Conclusion partielle et discussion

Dans ce paragraphe, nous avons décrit les alternatives à un encodage binaire des descripteurs au moyen du vocabulaire visuel. Nous avons posé un regard critique sur la complexité des procédés et mis en avant des méthodes d'encodage plurivoques adaptées à la grande échelle. Dans un second temps, nous avons introduit la problématique des bases de données d'apprentissage pour des classificateurs robustes face à la diversité des contenus du Web. Nous avons présenté une méthode permettant d'estimer automatiquement la position de la fenêtre englobante sur une base de données d'images d'entraînement.

Obj1.2.i : étudier les différentes stratégies d'encodages au moyen d'un vocabulaire visuel de moyenne taille.

Nous avons présenté les critiques formulées à l'endroit de la méthode d'encodage binaire. Nous avons alors présenté les alternatives utilisées dans la littérature et pointé des méthodes adaptées à la grande échelle le cas échéant (LLC).

Obj1.2.ii : Traiter le problème de l'achalandage d'histogrammes du à la présence d'arrière-plan.

Nous avons introduit une méthode basée sur la construction d'un arbre de clusters permettant d'estimer les zones d'une image étiquetée qui contiennent des parties de l'objet d'intérêt.

3.3 Classification d'images

Il reste à présent à étudier les méthodes de classification adaptées au scénario des vidéos de concert et à construire une méthode capable de tirer partie d'images d'entraînement à contenu riche annoté automatiquement via l'algorithme décrit plus haut.

3.3.1 Position du problème

Approches paramétriques versus approches non paramétriques. En classification, les approches paramétriques consistent à apprendre un modèle à partir d'un jeu de données d'apprentissage, permettant d'étiqueter des données inconnues. Ces approches nécessitent donc une phase d'apprentissage, en vue d'optimiser une fonction objectif dépendant des paramètres à estimer. Les SVMs sont des approches paramétriques discriminantes très populaires en classification d'objets, du fait des performances obtenues sur plusieurs bases de données de référence (Pascal du VOC ou image-net). En pratique, on a recours à une représentation des images au moyen d'un vocabulaire de grande dimension et à une SVM linéaire pour la

classification. Lorsque le nombre d'images est très élevé, l'apprentissage peut être facilement parallélisé de sorte que l'échelle n'est pas un défit majeur en terme d'apprentissage pour de telles méthodes.

L'alternative aux méthodes paramétriques consiste à utiliser une méthode basée sur le principe de la classification par le plus proche voisin. L'avantage de ces méthodes est qu'elles ne nécessitent pas d'apprentissage. De plus, elles sont robustes à l'augmentation du nombre de classes d'objets. étant donné une image inconnue, la classification est réalisée en une étape de recherche du plus proche voisin (plusieurs calculs de distance), tandis qu'une stratégie basée sur les SVMs impose de tester l'image avec chaque classificateur appris (un par classe d'objets).

Nous présentons ici une rapide étude comparée des performances des deux classificateurs sur le problème qui nous concerne (la reconnaissance de sept instruments sur des images difficiles). Le tableau 3.6 présente les résultats obtenus respectivement par un jeu de sept SVMs versus un classificateur par le plus proche voisin sur la base de données GoogleDB (sans arrière-plan), et le tableau 3.7 présente les résultats obtenus sur ImagenetDB (images complexes). La précision obtenue au moyen des SVMs linéaires est supérieure pour les deux bases de données. Par ailleurs, la précision et le rappel chutent pour les deux méthodes lorsque l'on passe de la base de données GoogleDB à la base de données ImageNetDB, du fait de l'augmentation de la variabilité intra-classe et de la complexification du contenu des images (multiples objets, arrière-plan). En termes de temps d'apprentissage, les SVMs sont plus lourdes à entraîner, notamment du fait de la nécessité d'apprendre les paramètres de biais et de coût de la fonction objectif. La classification par le plus proche voisin, elle, ne demande que le temps de construction de l'arbre de recherche (Tableau 3.8).

Tableau 3.6 Précision-recall pour le problème de classification binaire- SVM (L1R-L2LOSS-SVC) versus NN - Tain-350(GoogleDB)

	SVM		NN	
	précision	rappel	précision	rappel
guitares	0.9	1.0	0.72	0.8
batterie	1.0	1.0	0.75	0.3
harpes	0.7	0.78	0.26	0.6
accordéons	1.0	0.87	1.0	0.6
saxophones	0.89	0.89	0.78	0.7
trompettes	0.8	0.89	0.67	0.4
pianos	0.85	0.6	0.72	0.8
moyenne	0.87	0.86	0.7	0.6

Facteurs critiques. L'étude précédente met en évidence les avantages et les inconvénients

Tableau 3.7 Précision-recall pour le problème de classification binaire- SVM (L1R-L2LOSS-SVC) versus NN - Train-1400 (imageNetDB)

	SVM		NN	
	précision	rappel	précision	rappel
guitares	0.3	0.3	0.23	0.22
batteries	0.38	0.4	0.19	0.18
harpes	0.43	0.35	0.3	0.3
accordéons	0.44	0.44	0.4	0.36
saxophones	0.35	0.3	0.26	0.34
trompettes	0.37	0.33	0.45	0.46
pianos	0.40	0.45	0.31	0.3
moyenne	0.40	0.45	0.31	0.3

Tableau 3.8 Performances des SVMs vs. NN

	SVM		NN	
Base de données	t(s) train	t(ms) test	t(ms) train	t(ms) test
googleDB	>> 60	0.1	345	0.9
pascalDB	>> 60	2.3	4890	3.4
imageNetDB	>> 60	2.5	5980	3.9

des deux approches, par SVMs et par recherche du plus proche voisin. Quel que soit le degré d'optimisation, les SVMs souffrent lors de l'augmentation du nombre de classes d'objets d'intérêt car il est nécessaire d'apprendre un hyperplan séparant chaque classe des autres. En plus du problème du temps consacré à l'apprentissage dans ce scénario, l'aptitude d'un hyperplan à séparer une classe des autres diminue si le nombre de classes augmente. De plus, ce type de séparation linéaire ne gère pas les problèmes multinomiaux.

Le classificateur par le plus proche voisin quant à lui, souffre moins de l'augmentation du nombre de classes. Cependant, du fait du phénomène de dentelure de la frontière entre les classes, il s'avère moins robuste que les SVMs. Le classificateur par le plus proche voisin est donc très sensible aux points isolés. Choisir k-voisins plutôt qu'un seul ne résout pas le problème. Il s'agit en effet de prendre une décision étant donné les voisins retrouvés, ce qui n'est pas trivial. L'avantage des SVMs ici s'explique également par le fait que les SVMs apprennent un classificateur paramétrique tandis que le classificateur par le plus proche voisin utilise une distance élémentaire sans apprentissage des composantes importantes dans la représentation des images.

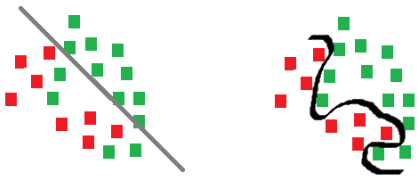

Figure 3.13 SVM (gauche) versus classification par le plus proche voisin (droite).

3.3.2 Renforcer le classificateur par le plus proche voisin

Approches hybrides. Les approches hybrides visent à combiner les deux philosophies présentées au paragraphe 3.3.1. L'idée consiste à paramétrer le classificateur par le plus proche voisin pour garantir une classification plus précise. On obtient de la sorte un classificateur robuste, adapté aux problèmes à grande échelle.

L'une des approches les plus illustratives de la motivation des approches hybrides est sans doute la méthode introduite par Zang et al. en 2006. (Zhang *et al.*, 2006), qui consiste à apprendre un SVM dynamiquement à partir des k-voisins retrouvés pour une image inconnue. La technique de Zang et al. consiste à estimer une frontière locale en optimisant la fonction objectif d'une SVM multiclasse non-linéaire 3.14. Elle vise à corriger l'erreur du classificateur due au phénomène de dentelure de la frontière de décision. La méthode est séduisante, mais s'avère décevante en pratique, notamment à cause de l'entrainement d'une SVM non linéaire sur un nombre réduit d'exemplaires (overfitting).

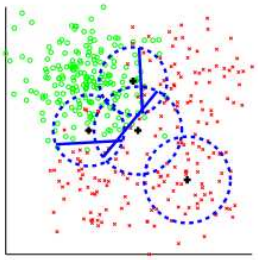

Figure 3.14 Représentation schématique de la méthode knnSVM (Zhang *et al.*, 2006).

Dans la revue de littérature (2.1), nous avons déjà évoqué les approches d'apprentissage de distance globales (LDA,LDE) et locales (LLDA,LLDE,LFR). Comme le soulignent Kim

et Kittler (Kim et Kittler, 2005a), les méthodes globales sont, de manière générale, très mal adaptées à la grande échelle et au caractère multinomial des données. Les méthodes d'apprentissage local, quant à elles sont fortement dépendantes du voisinage retrouvé pour une image inconnue. La question est alors, pour des images complexes, le voisinage retourné par un classificateur par le plus proche voisin est-il assez pertinent pour pratiquer l'apprentissage local ?

Arbres informés. L'apprentissage linéaire global semble trop grossier tandis que l'apprentissage local agit trop tard, après que des choix peu judicieux aient été réalisés au moyen de la distance L_2. En fait, il semble que sur des bases de données complexes, la distance euclidienne soit inefficace pour retrouver les voisins appropriés pour la classification. Ceci nous amène à considérer les méthodes supervisées de construction d'arbres de recherche, de type arbres de décision, revues au paragraphe 2.1 et 3.2. Ici, nous adoptons une approche semblable à celle de Yao et al. (Yao *et al.*, 2011b) en nous basant sur un arbre de décision discriminant. Dans leur article, Yao et al. construisent des arbres de décision binaires, dont les noeuds sont itérativement divisés au moyen d'une SVM linéaire. Pour mémoire, les noeuds sont constitués d'un ensemble de régions étiquetées. L'entraînement d'une SVM nécessite un jeu d'exemples bien choisis, qui conditionnent l'équation de l'hyperplan séparateur obtenu. Dans (Yao *et al.*, 2011b), le jeu d'exemple est choisi en assignant à chaque classe présente dans le noeud une étiquette binaire, de sorte que le problème d'apprentissage revient à apprendre la frontière entre deux ensembles de classes d'objets. La technique que nous décrivons ici diffère de celle de Yao et al. par le mode de scission des noeuds, ainsi que par un ensemble de mécanismes subtils, qui permettent de maintenir la qualité des jeux de données pour l'apprentissage des SVMs lors de la descente dans l'arbre.

3.3.3 Approche proposée : Cascade de SVMs pour la recherche du plus proche voisin (SVM Cascade for Nearset-Neighbor search SVMCNN)

Motivations

Le travail décrit en 3.3.2 nous amène à identifier trois facteurs clés limitant la précision des algorithmes sur les images issues de vidéos de concert. D'abord, la distance L_2 ne procure pas un voisinage assez précis, ce qui empêche de se reposer sur des méthodes d'apprentissage local. Ensuite, le passage d'images centrées sur l'objet à des scènes complexes impacte fortement la précision des algorithmes. Enfin, les techniques d'apprentissage globales ne gèrent pas la multinomialité. L'idée ici est de construire un classificateur adapté à la grande échelle qui soit robuste à ces trois phénomènes.

Au terme du travail décrit au paragraphe 3.2, nous disposons d'un ensemble de régions

encodées sous forme d'histogrammes éparpillés. à chaque région est associé un score proportionnel à la pertinence de la région pour représenter la classe associée à l'image dont est issue la région en question. En fixant un seuil par image, on peut constituer un jeu de données d'exemplaires positifs et négatifs par classe (figure 3.15). On dispose, par exemple, de régions représentatives de la classe saxophones, et de régions n'appartenant pas à cette classe.

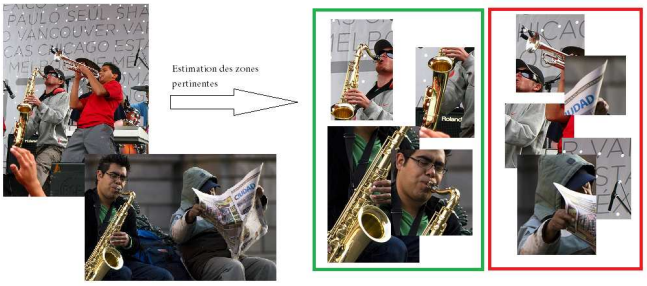

Figure 3.15 Base de donnée obtenue au terme de 3.2.3. (recangle vert : sax positif, rectangle rouge : non sax)

Définitions

On note N_I le nombre d'images d'entrainement, c_i la classe de l'image i $\in N_I$. H_i est l'ensemble des R_i régions construites sur l'image i. Pour mémoire, les régions sont soit des régions pertinentes au sens d'une classe donnée, soit de l'arrière-plan.

Définition 1 : La fonction évaluant si la region h_{ij} contient l'objet d'intérêt de classe c_k est définie par :

$$\forall i \in N_I \ \forall j \in R_i, \ b(h_{ij}, c_k) = 1 \text{ si } h_{ij} \text{ contient } c_k, \text{ 0 sinon.} \tag{3.28}$$

cette fonction résulte de l'apprentissage réalisé en 3.2.

Définition 2 : On dit qu'une classe $c \in C$ **appartient** à un noeud Nd si elle est suffisament représentée dans ce noeud. Ainsi, $\forall c \in C, c$ **appartient** à Nd si et seulement si :

$$rep(c, Nd) \geq \epsilon \tag{3.29}$$

Avec :

$$rep(c, Nd) = \frac{|c \cap Nd|}{|c|} \tag{3.30}$$

$$= \frac{\sum_{i \in N_I \cap c} (h_{ij} \in Nd) * b(h_{ij}, c_i)}{\sum_{i \in N_I \cap c} b(h_{ij}, c_i)} \tag{3.31}$$

Définition 3 : On dit que la classe **dominante** d'un noeud Nd est la classe c vérifiant :

$$c = max_c(pur(c, Nd)) \tag{3.32}$$

Avec :

$$pur(c, Nd) = \frac{|c \cap Nd|}{|Nd|} \tag{3.33}$$

$$= \frac{\sum_{i \in N_I \cap c} (h_{ij} \in Nd) * b(h_{ij}, c_i)}{\sum_{i \in N_I} (h_{ij} \in Nd) * b(h_{ij}, c_i)} \tag{3.34}$$

Définition 4 : On dit qu'un noeud est **pur** si il contient une seule classe au sens de la définition 1.

Définition 5 : On dit qu'un noeud Nd est ambigu si au moins 50% des régions de Nd ne contiennent pas la classe dominante de Nd.

Structure de l'arbre

L'arbre construit est un arbre binaire. à chaque noeud Nd, les données sont séparées de part et d'autre d'un hyperplan P_{Nd}, dont l'équation est estimée au moyen d'une SVM linéaire, entraînée sur deux ensembles bien choisis.

La division d'un noeud Nd a lieu si et seulement si le noeud Nd vérifie l'une des deux conditions suivantes :

1. le noeud Nd contient au moins deux classes.
2. le noeud Nd contient une classe et est ambigu.

Si les deux conditions précédentes sont violées, le noeud est une feuille. Si la feuille contient au moins une classe c, on procède à la séparation des régions informatives au sens de c, des régions ne contenant pas c. Si la feuille ne contient aucune classe, il s'agit d'une feuille d'arrière-plan.

Partition des noeuds

Dans chaque noeud Nd, une classe c_p est choisie parmi les classes appartenant au noeud. On choisi toujours la classe de représentativité minimale qui **appartient** au noeud, et en cas d'égalité, on tranche par un choix aléatoire. La classe c_p sert alors de pivot pour la partition du noeud. Les échantillons positifs ($h_{i,j} \in Nd$ et $b(h_{i,j}, c_i) = 1$), de cette classe contenus dans le noeud forment les exemplaires positifs $\{+\}$ pour l'entraînement de la SVM linéaire associée au noeud. Les exemplaires négatifs sont choisis aléatoirement dans l'ensemble $Nd - \{+\}$ de sorte à égaler en nombre les échantillons positifs. Le séparateur à marge maximale est calculé par descente de gradient pour séparer optimalement $\{+\}$ et $\{-\}$. Le contenu du noeud est alors réparti entre les deux enfants selon cet hyperplan (Fgure 3.16).

Comme l'illustre la figure 3.16, certaines classes sont susceptibles de couvrir la frontière de séparation. Ignorer ce phénomène reviendrait à pénaliser ces classes, pour lesquelles la frontière calculée n'est pas optimale. Soit N_{ga} et N_{dr} les noeuds gauche et droit issus du noeud parent. On défini donc τ le taux de séparation d'une classe $c \neq c_p$, comme :

$$\tau_c = min(\frac{|c \cap N_{ga}|}{|c \cap N_{ga}| + |c \cap N_{dr}|}, \frac{|c \cap N_{dr}|}{|c \cap N_{ga}| + |c \cap N_{dr}|}) \tag{3.35}$$

Ici, nous distinguons deux cas de figure.

- $\tau_c < \epsilon$: la classe c est majoritairement située d'un côté de l'hyperplan. On duplique donc les exemplaires envoyés vers le noeud minoritaire dans le noeud majoritaire. Ceci évite le phénomène d'essaimage des exemplaires.
- $\tau \geq \epsilon$: dans ce cas de figure, on ne duplique pas les éléments séparés.

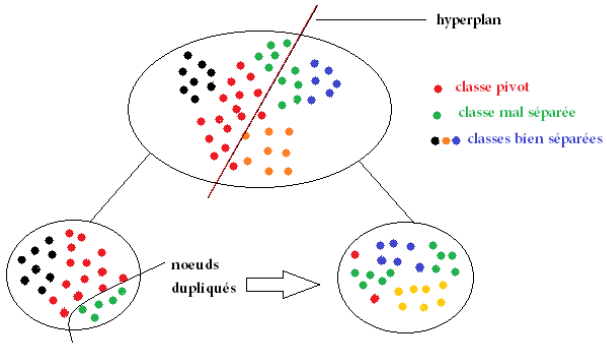

Figure 3.16 Construction de l'arbre SVMCNN

Désambiguation des feuilles et classification

Désambiguation des feuilles. En répétant le processus de scission duplication présenté plus haut, les noeuds s'épurent au fur et à mesure que l'on s'éloigne de la racine et des feuilles sont formées. Certaines ne contiennent que de l'arrière plan :

$$\forall h_{ij} \in f,\ b(h_{ij}, c_i) = 0 \tag{3.36}$$

Les autres contiennent une seule classe dominante c_d, qu'il s'agit d'apprendre à séparer des autres éléments présents dans la feuille. Ces éléments sont soit des régions pertinentes d'autres classes (bruit), soit de l'arrière-plan. Quoi qu'il en soit, les éléments qui parviennent dans une feuille sont très proches du contenu de la feuille et ne peuvent être distingués de ce contenu par un classificateur linéaire simple tel que les SVMs. Aussi, avons-nous recours à un modèle plus robuste qu'une SVM linéaire. Après réduction de la dimensionnalité via une PCA, nous apprenons une mixture de k gaussiènes diagonales, de sorte que la classification d'une image inconnue q parvenant dans la feuille est donnée par :

$$P(C_d) = \sum_k P(q,k) = \sum_k P(q|k)P(k) \tag{3.37}$$

Résultats expérimentaux

La méthode décrite en 3.3.3 est appliquée au problème de classification binaire sur la base de données ImageNetDB. Par rapport à une approche par SVMs simple, on obtient une hausse de 14% de la précision et de 8% du rappel, soit un gain de 10% du F1-score.

Tableau 3.9 F1-score pour notre méthode (SVMCNN) et pour des SVMs linéaires (LSVM)

	SVMCNN			LSVM		
classe	précision	rappel	F	précision	rappel	F
guitares	0.39	0.64	0.48	0.3	0.3	0.3
batteries	0.37	0.52	0.42	0.38	0.4	0.39
harpes	0.2	0.41	0.26	0.43	0.35	0.38
accordéons	0.46	0.6	0.52	0.44	0.44	0.44
saxophones	0.43	0.41	0.42	0.35	0.3	0.33
trompettes	0.8	0.4	0.6	0.37	0.33	0.35
pianos	1.0	0.15	0.26	0.4	0.45	0.43
moyenne	0.52	0.44	0.47	0.38	0.36	0.37

Les résultats obtenus ici sont difficiles à comparer avec l'état de l'art sur image-net (LSVRC) car la procédure d'évaluation pour cette base de données est particulière et ne repose

pas sur une stricte évaluation de la précision ou du rappel. Les gagnants du challenge LSVRC 2011 (Perronnin et Sanchez, 2011) par exemple, calculent le score obtenu en comptabilisant le nombre de fois que l'étiquette correcte figure dans les cinq premières prédictions. Ici, avec sept classes, ce type d'évaluation n'a pas de sens. Notons tout de même la différence de stratégie : Perronin et al. utilisent des vecteurs de Fisher de 130K dimensions, tandis que la représentation que nous utilisons est plus compacte (1K dimensions). Dans notre cas, l'apprentissage vient combler la perte d'information due à la quantification, tandis que pour Perronin, c'est au niveau de l'encodage que le travail est réalisé.

3.3.4 Conclusion partielle et discussion

Dans ce paragraphe, nous avons comparé les performances de deux classificateurs adaptés à la grande échelle : les SVMs linéaires et le classificateur par le plus proche voisin. Nous avons en lumière leurs points forts et leurs limites et avons proposé une stratégie alternative, dont nous avons montré empiriquement qu'elle surpasse les résultats obtenus avec l'une et l'autre méthode.

Obj1.3.i : évaluer les performances d'un classificateur par le plus proche voisin en présence d'un nombre suffisant d'instances. Au paragraphe 3.3.1, nous avons mesuré les performances du classificateur par le plus proche voisin et souligné son imprécision, en dépit du nombre d'instances disponibles pour l'apprentissage. Nous avons en particulier expliqué le phénomène de dentelure de la frontière entre les classes en présence d'un échantillonnage fini. De plus, nous avons montré qu'un apprentissage local est insuffisant pour des scènes riches encodées avec un petit vocabulaire.

Obj1.3.ii : Proposer des solutions pour rendre plus robuste le classificateur par le plus proche voisin. Nous avons alors proposé une approche basée sur un arbre de décision amélioré, dont les noeuds sont des SVMs linéaires. Dans la partie 3.3.3, nous avons montré empiriquement l'avantage, à encodage égal, de l'usage de notre méthode plutôt que des SVMs linéaires élémentaires.

3.4 Conclusion du chapitre

L'élaboration d'un détecteur d'objets adapté au contenu des vidéos de concert suppose de relever plusieurs défis, notamment la robustesse aux grandes échelles, la gestion de la variabilité intra-classe et la capacité à apprendre sur des images non idéales.

La dimension de la représentation des images est un paramètre critique pour le passage à la grande échelle. Pourtant, l'état de l'art en reconnaissance d'objets tend à utiliser des vecteurs de dimension de plus en plus élevée, quitte à avoir recours à des structures et

techniques complexes pour leur manipulation. Ici, nous avons suggéré qu'une représentation de moyenne dimension est suffisante pour le problème de classification si l'on a recours à un classificateur plus riche au moment de l'apprentissage.

La plupart des algorithmes de détection d'objets ne prennent que mal en compte la multinomialité des données : les SVMs linéaires la gèrent très mal et les modèles en mixture - d'exponentielle ou d'autre chose, sont dépendants d'un paramètre arbitraire. Le classificateur présenté dans ce chapitre gère dynamiquement le problème de la multinomialité.

Enfin, lorsque l'on sort des cas d'école, la qualité de l'apprentissage dépend fortement de la représentativité des données d'apprentissage. Or, obtenir des données en nombre suffisant et variées n'est pas une démarche aisée. Ce type de préoccupation est central pour la recherche en traitement d'image à l'échelle du Web. Aussi, avons-nous consacré nos efforts à produire un algorithme capable de simplifier la procédure d'étiquetage de données en automatisant le calcul des boîtes englobantes.

En somme, nous avons travaillé à rendre les algorithmes de référence en reconnaissance d'objets un peu plus applicables au contexte du Web et plus spécifiquement, aux vidéos de concert. Dans le chapitre suivant, nous poursuivons le même objectif, en nous plaçant du point de vue de la recherche en traitement de visages.

CHAPITRE 4

TECHNIQUES DE RECONNAISSANCE DE VISAGES ADAPTEES AUX VIDEOS DE CONCERT

La finalité de l'apprentissage en reconnaissance de visage n'est plus, comme en reconnaissance d'objets, de reconnaître ce qui caractérise un groupe d'entités, mais ce qui caractérise une entité spécifique, c'est-à-dire une personne. Ceci suppose de mettre en jeu des techniques d'apprentissage différentes. De plus, les contraintes associées à l'apprentissage sont d'un autre ordre que les contraintes identifiées au chapitre précédent. Ainsi, en classification de visages, le défi n'est pas tant de traiter de scènes complexes, mais plutôt de gérer la variabilité des poses et des expressions d'un individu. Il est également nécessaire de prendre en compte le caractère limité du nombre des instances d'apprentissage disponibles, du fait du cadre législatif sévère encadrant les portraits humains.

En reconnaissance de visages, il s'agit de distinguer les visages selon l'identité des personnes. En pratique, la recherche se concentre sur l'apprentissage de fonctions permettant de séparer deux ensembles dans l'espace des distances entre les images : l'ensemble des distances entre deux images d'un même individu, et l'ensemble des distances entre deux images d'individus différents. On parle de vérification. Pour séparer au mieux ces deux ensembles, la pratique consiste à projeter les images dans un espace approprié, tel que dans cet espace les deux ensembles sont bien séparés. Aujourd'hui, les méthodes de vérification obtiennent d'excellents résultats sur des bases de données reflétant la réalité des contenus du web. Cependant, le passage à la classification et, dans un second temps, à la grande échelle, nécessaire pour l'application pratique aux vidéos de concert, restent deux défis ouverts en 2012.

Dans ce chapitre, nous présentons dans un premier temps les méthodes classiques d'encodage et de comparaison des visages. Nous posons la question du passage de la vérification à la classification et proposons des pistes pour le renforcement des classificateurs élémentaires. Ensuite, nous examinons deux facteurs critiques qui limitent les performances des algorithmes de classification, à savoir les variations de pose et le nombre réduit des exemplaires d'apprentissage par individu. Nous terminons en examinant le scénario de la classification à grande échelle et en proposant une méthodologie adaptée à ce type de défi.

De la vérification à la classification. Dans le contexte du web, les visages peuvent être extraits d'images complexes via le classificateur en cascade de Viola et Jones (Viola et Jones, 2001). Il s'agit alors d'identifier les zones discriminantes des portraits qui permettent de distinguer les individus. Les changements dans la pose des sujets empêchent la comparaison

directe, car les visages ne sont pas alignés. C'est pourquoi, on procède à un travail d'alignement des visages. Suite à cette étape cruciale de standardisation, il est commun d'apprendre une fonction dite de *vérification*, destinée à identifier les paires d'individus identiques et les paires d'individus différents. Cette technique est aujourd'hui hautement maîtrisée. Cependant, disposer d'une fonction de vérification robuste n'implique en rien que la classification des visages l'est tout autant. Quel est l'ordre de grandeur de la précision d'un système de classification d'individu sur des portraits réalistes ? Peut-on améliorer ces performances ? Cette question est abordée au paragraphe 4.1.

Variation de pose et du nombre d'exemplaires d'apprentissage. Comme nous l'avons mentionné plus haut, les variations de pose peuvent être gérées via une procédure d'alignement des visages. Cependant, ce procédé reste imparfait. Nous proposons donc d'évaluer l'apport d'une procédure d'apprentissage prenant explicitement en compte la pose des visages. Dans un second temps, nous discutons d'un autre facteur problématique qui impacte fortement les performances des classificateurs, à savoir le nombre d'exemplaires d'apprentissage disponibles. Comment une stratégie prenant en compte la pose des individus peut-elle aider à distinguer les visages ? Comment se procurer des visages supplémentaires pour l'apprentissage et quel est l'impact d'une telle stratégie sur la méthode de classification ?

Vers la très grande échelle. Le nombre d'instances d'apprentissage conditionne la précision d'un classificateur. Or, lorsqu'on parle de visages, la disponibilité des ressources étiquetées devient problématique. S'il est possible d'utiliser des moteurs de recherche pour obtenir des visages de personnalités, les exemplaires obtenus sont quelquefois erronés. Quelles stratégies sont envisageables pour pratiquer la classification de visages à très grande échelle sans souffrir du manque d'exemplaires d'apprentissage ? Nous abordons ce point dans la troisième partie de ce chapitre.

Dans ce chapitre, nos expériences sont réalisées sur la base de données Labelled Faces in the Wild (LFW), que nous avons choisi pour son réalisme en termes de variation dans la pose et l'expression des sujets.

4.1 De la vérification à la classification

4.1.1 Représentation et comparaison des visages

Détection de visages, alignement, et représentation. L'algorithme de Viola et Jones (Viola et Jones, 2001) permet de réaliser des détections robustes sur les images à contenu riche. Cet algorithme a recours à une série de classificateurs en cascade (cascades de Haar), ce qui rend la détection très rapide. De plus, il présente une robustesse très respectable pour son efficacité, en particulier lorsqu'associé à une vérification via un second détecteur (d'oeuil, nez,

ou bouche), ce qui en fait un des standards en matière de détection de visages. Les visages extraits par un tel détecteur présentent une grande variabilité en terme de pose et d'échelle. Aussi, procède t'on à un travail de mise à l'échelle et d'alignement sur un axe commun de $\mathbb{R}^2$, $\mathbb{R}^3$. Ce travail d'alignement est rendu possible par la régularité de la morphologie des visages, qui permet d'estimer facilement la matrice de similitude pour l'alignement.

Plus spécifiquement, on pratique la détection de points clés sur le visage (les yeux, le nez et les coins de la bouche) et on estime la transformation permettant d'aligner ces points avec les points correspondants sur une image de référence. En 2D, le passage des coordonnées d'un point (x,y) dans le système de coordonnées propres à l'image à des coordonnées dans le référentiel de référence peut être estimé par une similitude, soit :

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} m1 & -m2 \\ m2 & m1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \tag{4.1}$$

L'apprentissage de la transformation pour l'alignement d'une image consiste donc à estimer les paramètres $(m1, m2, t_x, t_y)$. Si $(x1, y1), (x2, y2), ..(xn, yn)$ sont n points clé détectés sur l'image à aligner, et $(u1, v1), (u2, v2), ...(un, vn)$ les points correspondant sur le visage de référence, alors on peut écrire le système d'équations suivant :

$$\begin{bmatrix} x1 & y1 & 1 & 0 \\ y1 & x1 & 0 & 1 \\ x2 & y2 & 1 & 0 \\ y2 & x2 & 0 & 1 \\ ... & ... & ... & ... \\ xn & yn & 1 & 0 \\ yn & xn & 0 & 1 \end{bmatrix} \begin{bmatrix} m1 \\ m2 \\ t_x \\ t_y \end{bmatrix} = \begin{bmatrix} u1 \\ v1 \\ u2 \\ v2 \\ ... \\ un \end{bmatrix} \tag{4.2}$$

Sur la base de données LFW, le meilleur aligneur est un aligneur commercial (Wolf *et al.*, 2009b), suivi de près par l'aligneur de Hasan et al. (M. K. Hasan, 2011), que nous utilisons dans ce mémoire. Une fois alignés, les visages sont décrits au moyen de descripteurs SIFTs, HOGs, ou LBPs, voire une combinaison des ces descripteurs. Dans le cadre de ce travail, nous utilisons des visages encodés par LBPs, fréquement utilisés sur LFW.

Comparaison des visages. Après encodage par LBP, la représentation d'un visage est un vecteur de très grande dimension (environs 7,000). L'analyse en composantes principales (PCA) est une méthode de réduction de la dimensionnalité non supervisée, qui, si appliquée trop radicalement sur l'ensemble d'apprentissage, est susceptible d'éliminer trop d'information et de nuire à l'entraînement. Aussi, lui préfère-t-on des méthodes de réduction de la dimensionnalité plus informées.

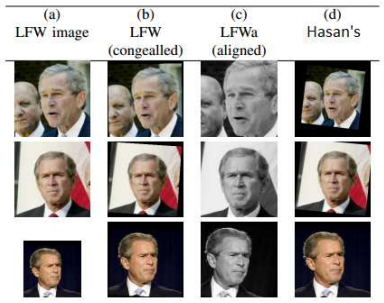

Figure 4.1 Alignement d'images par Hasan et al. (M. K. Hasan, 2011)

La technique de réduction de la dimensionnalité supervisée la plus célèbre en traitement de visages est sans conteste l'analyse discriminante linéaire de Fisher (Fisher-LDA Belhumeur *et al.* (1997)). Dans sa version non kernélisée, la technique de Fisher LDA consiste à maximiser le ratio de la variance inter-classe par la variance intra-classe :

$$f(w) = \frac{w'\Sigma_b w}{w'\Sigma_i w} \tag{4.3}$$

avec $\Sigma_b = \sum_c (\nu_c - \bar{x})(\nu_c - \bar{x})'$ et $\Sigma_i = \sum_c \sum_{i \in c} (\nu_c - x_i)(\nu_c - x_i)'$.

On peut montrer que les vecteurs propres généralisés associés à l'équation $\Sigma_b w = \lambda \Sigma_i w$ sont solution de l'équation 4.3. En effet, (Belhumeur *et al.*, 1997), une des propriété de l'équation 4.3 est que f(w) est invariante aux multiplication de w par un scalaire. Ainsi, on peut choisir w tel que $w'\Sigma_i w = 1$.

L'équation 4.3 peut donc être reformulée en :

$$f'(w) = w'\Sigma_b w \text{ s.c } w'\Sigma_i w = 1 \tag{4.4}$$

Le lagrangien associé à f' est alors :

$$L = -w'\Sigma_b w + \lambda(w'\Sigma_i w - 1) \tag{4.5}$$

Sous les conditions de Karush Kuhn Tucker (KKT), la solution vérifie donc :

$$\Sigma_b w = \lambda \Sigma_i w \tag{4.6}$$

w est donc solution du problème aux vecteurs propres généralisés défini par l'équation 4.6.

Bien qu'il ait démontré son succès sur de nombreux problèmes, le modèle de Fisher repose sur l'hypothèse forte que les classes ont des moyennes distinctes. De plus, la fonction objectif est basée sur des statistiques fortes concernant chaque classe d'individus. Or, dans le cadre d'une application pratique à grande échelle, le nombre d'exemplaires d'apprentissage est souvent très réduit (≤ 5) de sorte que l'utilisation de paramètres tels que la moyenne ou la variance d'une classe peut être contestée.

Ce problème concerne en fait tous les modèles d'apprentissage supervisé basés sur un paramétrage par classe. C'est pourquoi, sur des bases de données de de visages à grande échelle telle que LFW, on préfère avoir recours à des modèles visant à optimiser la distance entre les paires de visages similaires et les paires de visages différents, sans spécialisation par identité. La LDE (Chen *et al.*, 2005) appartient à cette classe de modèles. Elle découle directement du souci de supprimer la dépendance de la fonction objectif de la LDA aux classes d'individus. Ainsi, dans le cadre de la LDE, la fonction objectif est formulée comme :

$$f(w) = \frac{w'Aw}{w'Bw} \tag{4.7}$$

avec $A = \sum_{-}(x_i - x_j)(x_i - x_j)'$ et $B = \sum_{+}(x_i - x_j)(x_i - x_j)'$.

La fonction objectif de la LDE peut être vue comme un rapport de *variances* non paramétriques. Une fois de plus, on peut montrer que les vecteurs propres généralisés associés à l'équation $Aw = \lambda Bw$ sont solution de l'équation 4.12. La LDE peut être vue comme une version de Fisher non paramétrique.

Dans la même veine, la méthode d'apprentissage de distance en cosinus (Cosine Similarity Metric Learning, CSML) vise à optimiser l'écart entre les paires de visages similaires et les paires de visages différents dans l'espace des similarités :

$$CS(x, A, y) = \frac{x'A'Ay}{\|Ax\|\|Ay\|} \tag{4.8}$$

Plus spécifiquement, la procédure CSML consiste à apprendre la matrice A minimisant :

$$F(A) = \sum_{+} CS(x, A, y) - \alpha \sum_{-} CS(x, A, y) + \beta \left\| A - A_o \right\| \tag{4.9}$$

avec $\{+\}$ un ensemble de paires d'images appartenant au même individu, et $\{-\}$ un ensemble de paires d'images appartenant à des individus différents. Le coefficient α est un coefficient de normalisation du poids des ensembles $\{+\}$ et $\{-\}$, β est la pénalité associée à la distorsion de A par rapport à A_o, où A_o est initialisée via une technique d'analyse en composantes principales (Principal Components Analysis, PCA).

N'Guyen et al. proposent d'optimiser $F(A)$ par descente de gradient (Nguyen et Bai, 2010) :

$$\frac{\partial F}{\partial A} = \sum_{+} \frac{\partial CS(x, A, y)}{\partial A} - \alpha \sum_{-} \frac{\partial CS(x, A, y)}{\partial A} + \beta \frac{\partial \left\| A - A_o \right\|}{\partial A} \tag{4.10}$$

En termes de résultats, sur la base de données LFW, la méthode d'apprentissage par CSML permet d'obtenir 85% de prédictions correctes en vérification (*restricted settings* Nguyen et Bai (2010)), loin devant la Fisher LDA à 70% (Turk et Pentland., 1991). à notre connaissance, la LDE n'a pas été testée sur la base de données LFW. En revanche, elle a été notamment appliquée avec succès en classification (Kim et Kittler, 2005b), sous sa version locale (LLDE), sur la base de données CMU Pie, qui compte 68 sujets présentés dans des conditions de pose et d'éclairage variées.

4.1.2 De la vérification à la classification

La tâche de vérification consiste à déterminer si deux visages appartiennent au même individu (paire positive) ou à deux individus distincts (paire négative). La tâche de classification, elle, consiste à déterminer l'identité d'un visage inconnu. Aujourd'hui, sur la base de données LFW, on rapporte des taux d'erreur en vérification inférieurs à 10%. Malheureusement, comme nous le verrons plus loin, disposer d'un système performant de vérification n'implique pas que l'on possède un classificateur robuste.

Dans le cadre de la CSML, la fonction de vérification consiste en une fonction booléenne comparant la similarité en cosinus à un seuil optimisé pour séparer les paires d'images positives des paires d'images négatives. Dotés d'une telle fonction, on peut alors concevoir un classificateur trivial, consistant à comparer un visage inconnu à des exemplaires de chaque identité potentielle. La question que nous posons ici concerne la dégradation de la précision lors du passage de la tâche de vérification à la tâche de classification.

La classification de visages sur la base de données LFW n'a été que rarement essayée, sans doute du fait du nombre réduit de sujets contenant plus de vingt images, seuil en deça duquel les techniques classiques d'apprentissage supervisé s'effondrent. Rim et al. proposent de procéder à la classification des cinquante individus de LFW les plus dotés en images (Rim et Pal, 2011). La classification est réalisée au moyen d'un classificateur à marge maximale innovant, permettant d'atteindre une précision en classification de l'ordre de 80% pour les cinquante sujets. Ce sous-ensemble représente 1% environ des individus de la base de données LFW. La portée de la méthode reste donc limitée, car elle s'appuie sur le nombre important d'instances d'apprentissage disponibles pour quelques membres de LFW. Or, un tel nombre d'images d'apprentissage n'est pas représentatif de la distribution des images par individus

Figure 4.2 Exemple de paires de visages sur LFW

disponibles en ligne. Guillaumin et al. (Guillaumin *et al.*, 2009) proposent une amélioration de la méthode de classification par LMNN (*Large Margin Nearest Neighbor*, Weinberger *et al.* (2005)) appliquée à la tâche de vérification. Ils se proposent d'étendre le processus en classification, mais, la méthode requérant un nombre suffisant d'instances par individus, ils se cantonnent à dix-sept sujets.

Plus proche de notre projet, le travail de Wolf et al. (Wolf *et al.*, 2011) adresse le problème de classification des 610 sujets possédant plus de 3 images dans LFW. La classification est réalisée au moyen de SVMs multiclasses et le problème des individus possédant seulement quelques images est résolu via le recours à des images supplémentaires issues du web. Fort de ces images additionnelles, Wolf et al. obtiennent 45% de précision pour la classification de 610 individus. Cependant, sans ces images, sur LFW, la précision n'est que de 30%.

Ici, nous nous restreignons aux 1680 personnes possédant au moins deux images dans LFW. Nous séparons alors les images disponibles en un ensemble d'entrainement X et un ensemble de test Q. Pour ce faire, nous procédons à la division aléatoire du jeu d'images disponibles pour chaque identité en deux sous-ensembles égaux (modulo la parité). Nous procédons alors à la classification de chaque élément de Q par recherche de son plus proche voisin dans X, au sens de la CS. Ainsi, une image inconnue $q \in Q$ se voit attribuer l'identité $Y(x)$, tel que x vérifie :

$$x = max_{x \in X}(CS(q, x)) \tag{4.11}$$

Dans ce scénario, la précision observée (Tableau 4.4) est très inférieure aux résultats obtenus en vérification sur la base de données LFW avec la même méthode. On observe ici l'écart entre le problème de classification et de vérification.

La perte de précision lors du passage de la vérification à la tâche de classification n'est pas

un résultat surprenant. En vérification, chaque visage n'est comparé qu'à un sous-ensemble de l'ensemble d'apprentissage que nous avons utilisé ici. En choisissant la classification triviale par le plus proche voisin, nous tranchons de manière brute entre les images incertaines associées à un visage par la fonction de vérification, apprise sur un ensemble trop vaste pour trancher proprement entre les images voisines. Cette observation suggère qu'on aurait tout à gagner à pratiquer une identification locale, comme proposé par Guillaumin. Cependant, le nombre d'instances d'apprentissage disponibles pour chaque individu rend difficile l'apprentissage local supervisé.

Tableau 4.1 De la vérification à la classification (CS + force brute)

Base de données	Nb identités	|X|	|Q|	t(s)	précision
LFW	1680	4858	4308	21	0.32

Pour mémoire, Wolf obtient 45%, mais pour 610 sujets seulement. Nous considérons ici 1680 sujets.

4.1.3 Vers une meilleure séparation des visages dans l'espace des distances

La fonction objectif (équation 4.9) proposée par N'Guyen et al. (Nguyen et Bai, 2010) vise à séparer l'ensemble des paires de visages semblables (+) de l'ensemble des paires de visages différents (-) en optimisant l'écart des moyennes de deux ensembles dans l'espace des distances. En d'autres termes, N'Guyen et al. minimisent la distance moyenne associée aux paires de visages de l'ensemble (+) et maximisent la distance moyenne associée aux paires de visages de l'ensemble (-). On peut alors se poser la question de la variance de ces ensembles. En effet, si les moyennes s'éloignent, mais que les variances augmentent, il n'est pas évident que la séparation des deux ensembles soit améliorée par l'apprentissage de distances proposé par N'Guyen et al.

Or, la notion de variance est justement au coeur de la technique de Fisher, et de manière moins évidente, de la stratégie de projection linéaire discriminante (LDE) (Chen *et al.*, 2005). Pour mémoire, la fonction objectif de la LDE proposée par Chen et al. vise à maximiser le rapport A/B, avec :

$$A = \sum_{-}(x_i - x_j)(x_i - x_j)' \text{ et } B = \sum_{+}(x_i - x_j)(x_i - x_j)'. \tag{4.12}$$

Il est donc raisonnable de songer que CSML et LDE sont des techniques complémentaires. Nous proposons ici d'appliquer la LDE après CSML. Une fois les vecteurs projetés dans l'espace de la similarité en cosinus (équation 4.8), nous construisons les matrices A et B

(équation 4.12) sur l'ensemble d'images d'apprentissage utilisé pour le CSML. Nous résolvons ensuite le problème aux vecteurs propres généralisés solutions de l'équation 4.12, ce qui est facile puisque B est symétrique (en général définie) positive. La solution au problème consiste en les m vecteurs propres généralisés associés aux m plus grandes valeurs propres solution du problème (m la dimension de l'espace de projection).

En projetant à nouveau les vecteurs au moyen de la matrice de projection obtenue, on observe alors un gain en précision de 6%. De plus, en pratiquant la régularisation des vecteurs solution par la racine carrée de la valeur propre associée, on observe un gain en précision supplémentaire de 2% (Tableau 4.2).

$$x\ (7080\ dim) \overset{PCA}{\rightarrow} x_{PCA}\ (500\ dim) \overset{CS}{\rightarrow} x_{CS}\ (200\ dim) \overset{LDE}{\rightarrow} x_{LDE}\ (200\ dim) \tag{4.13}$$

Tableau 4.2 LDE after CSML

Base de données	dimension	dimension reduite	LDE	précision
LFW	200	200	non	0.32
LFW	200	200	oui	0.38
LFW	200	200	oui+regularisation	0.40

4.1.4 Autres facteurs critiques

Au-delà de la fonction objectif, quels sont les facteurs qui pénalisent les performances obtenues ? D'abord, on peut noter que le nombre d'instances disponibles pour l'apprentissage a un impact important. En effet, si l'on observe la précision obtenue par identité, on note une claire corrélation entre la précision et le nombre d'instances d'apprentissage (Figure 4.3). On peut alors songer, comme Wolf et al (Wolf *et al.*, 2011), que s'approprier des images supplémentaires sur le web dans le cadre d'une procédure d'apprentissage semi-supervisé permettra d'améliorer la précision des résultats.

Ensuite, si l'on observe la répartition de la distance associée aux paires d'individus sous la CS (Figure 4.4), on note que la séparation semble meilleure pour les paires d'individus (vue de face, vue de face) que pour les paires (vue de face, vue de profil). La figure 4.4 montre à gauche la répartition des distances associées aux paires d'images (face, face), à droite la répartition des distances associées aux paires (face, profil). En vert, sont représentées les paires d'individus identiques, en bleu, les paires d'individus différents. On constate que pour les paires (profil, face), le chevauchement des histogrammes semble plus important.

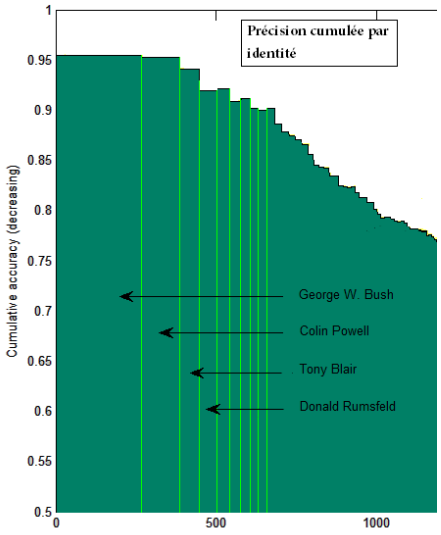

Figure 4.3 Influence du nombre d'instances d'apprentissage sur la précision. Précision cumulée calculée sur les individus triés par ordre décroissant d'images disponibles

Cet observation motive une approche d'apprentissage spécifique pour chaque combinaison de poses.

4.1.5 Discussion et conclusion partielle

Aujourd'hui, il existe des méthodes robustes de détection de visages sur des images à contenu complexe, dont la robustesse s'explique notamment par la constance dans la morphologie des visages. Le défi aujourd'hui consiste à identifier les visages. Si les méthodes de vérification se voient gratifier d'excellents scores sur la base de données LFW, le passage trivial de la tâche de vérification à la tâche de classification s'accompagne d'une perte de précision importante. Cette chute de la précision provient sans doute du sous échantillonnage (nécessaire) de l'espace des paires d'individus pour de l'apprentissage de fonctions de vérification. Quoi qu'il en soit, si les résultats obtenus en vérification peuvent laisser penser que la recherche en reconnaissance de visages est un domaine mûr et saturé, il reste encore une marge de progression importante pour la recherche en classification des individus.

Obj2.1.i : étudier les modes d'encodage et de comparaison des visages.

Dans ce paragraphe, nous avons décrit les techniques élémentaires d'extraction, d'alignement et de représentation des visages. Nous avons également présenté les stratégies sur lesquelles se basent les algorithmes de vérification constituant l'état de l'art aujourd'hui.

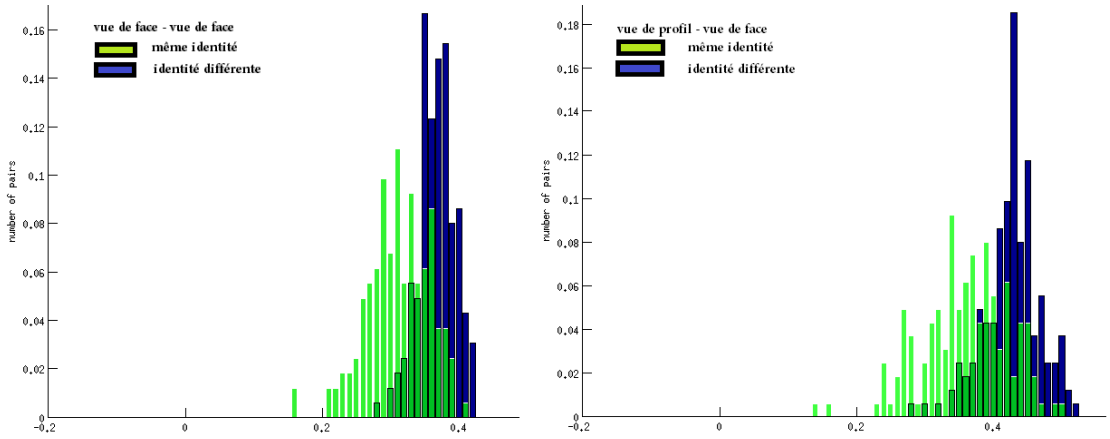

Figure 4.4 Influence de la pose des individus sur la précision de la CS. A gauche des paires (face,face), à droite, des paires de (face,profil). Abscisse : 1-CS, Ordonnée : nombre de paires

Obj2.1.ii : Proposer et évaluer une stratégie de classification de visages.

Dans un second temps, nous avons pratiqué une évaluation de la perte de précision lors du passage de la tâche de vérification à la tâche de classification. Nous avons expliqué ce phénomène et souligné plusieurs pistes de progression pour la recherche en classification des individus. En particulier, nous avons montré que le nombre d'instances disponibles pour l'apprentissage a un impact conséquent sur l'aptitude d'un classificateur à reconnaître un individu. Nous avons également mis en évidence l'influence des variations de la pose des individus sur l'efficacité de l'apprentissage CSML. Dans le prochain paragraphe, nous discutons de la viabilité et de la mise en oeuvre de ces deux pistes d'amélioration.

4.2 Renforcer la classification

4.2.1 Prise en compte de la pose

L'alignement des visages ne supprime qu'en partie le biais dû à la pose des individus sur les images. Lorsque l'écart de pose est trop extrême (profil, vue de face), on peut se demander s'il est même légitime d'appliquer le procédé. Ici, nous discutons de la mise en oeuvre d'une stratégie prenant explicitement en compte la pose des individus pour la classification.

La pose, entre autres facteurs, a notamment été identifiée comme une nuisance pour

les algorithmes de comparaison de visages par Yin et al. (Yin *et al.*, 2011), qui soulignent notamment qu'un trop grand écart dans la pose est susceptible de dominer les différences subtiles entres deux individus distincts photographiés dans les mêmes conditions. Dans (Yin *et al.*, 2011), ils gèrent le problème de la pose via une méthode basée sur le recours à de données externes à LFW. Leur idée consiste à ne réaliser de vérification que pour des visages photographiés dans des conditions similaires de pose et d'illumination. Dans le cas où deux visages (V1,V2) présentent des conditions trop différentes d'illumination ou de pose, ils ont recours à une base de données externe pour prédire un visage V3 associé à V2, qui présente des conditions de prise de vue similaires à celles de V1. La vérification est alors pratiquée entre V1 et V3, plus propres à être comparés. Sur LFW, la précision moyenne d'une telle méthode dépasse les 90% (*unrestricted settings*).

La stratégie proposée par Hasan et Pal (M. K. Hasan, 2011), à la différence de celle de Yin, prend en compte la pose de façon plus explicite. Ainsi, Hasan et Pal distinguent chaque étape de la procédure de traitement des visages selon la pose, de l'alignement à la vérification. Plus spécifiquement, ils pratiquent un apprentissage de la CS optimisé pour chaque combinaison de poses : (face,face), (face,profil), (profil,profil). Aucune donnée externe à LFW n'est requise. Ils montrent (Hasan *et al.*, 2012) que par ce procédé, le score de vérification peut être amélioré d'environs 1.5% par rapport à un apprentissage sans distinction sur la pose. La précision moyenne obtenue dépasse également les 90% (*restricted settings*). Nous nous intéressons ici à l'intégration de cette stratégie à un scénario de classification.

Hasan et Pal proposent une méthode capable de distinguer avec quasi certitude cinq types de poses (Figure 4.5). Leur méthode (M. K. Hasan, 2011) est entraînée sur des images de la base de données PUT [1] encodées au moyen de descripteurs locaux (HOG). Le taux d'erreur de leur algorithme d'estimation de pose est de l'ordre de 4.5%. Ici, nous ne retenons que trois des cinq poses présentées en Figure 4.3. pour simplifier le procédé. Par la suite, nous notons C = vue de font, R = profil gauche, L = profil droit.

La distinction selon la pose dans le cadre de la CSML consiste à apprendre une métrique spécifique pour la comparaison de chaque configuration : (CC,CR,CL,RR,LL,RL). Dans le cadre de ce travail, nous utilisons les visages de la base de données LFW, aimablement encodés, alignés et annotés de leur pose par K. Hasan. Nous utilisons également les matrices de projection apprises par K. Hasan par optimisation du la fonction objectif de N'Guyen et al. (équation 4.10), pour chaque sous-ensemble de paires (CC,CR,CL,RR,LL,RL).

Dans le scénario d'une classification basée sur la recherche du plus proche voisin au sens de la CS, distinguer les cas selon la pose implique de rechercher non pas un mais trois voisins pour classifier un visage inconnu. Si on note q_P une requête de pose $P \in (C, L, R)$, r_P une

1. https ://biometrics.cie.put.poznan.pl/

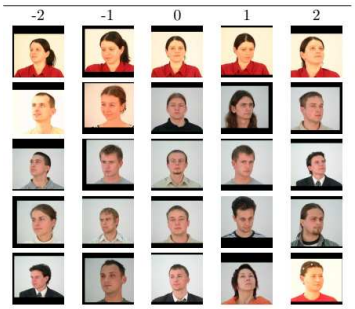

Figure 4.5 Les 5 classes de poses définies par M. K. Hasan (2011)

image de l'ensemble d'entraînement, de pose $P \in (C, L, R)$, on envisage les configurations suivantes : (q_C, r_C), (q_C, r_L), (q_C, r_R), (q_R, r_C), (q_R, r_R), (q_R, r_L), (q_L, r_C), (q_L, r_R), et (q_L, r_L). Par exemple, si la pose P de la requête est une vue de face (P=C), on trouvera un voisin de q_C parmi les visages vus de face, un voisin de q_C parmi les visages de profil droit, et un voisin de q_C parmi les visages de profil gauche. Pour la recherche de chaque voisin, on utilisera une projection dans un espace adapté : une projection dans l'espace des comparaisons (C, C), une projection dans l'espace des comparaisons (C, R) et une projection dans l'espace des comparaisons (C, L).

La question est alors, comment trancher parmi les trois identités retrouvées ? Pour résoudre ce problème, nous proposons d'apprendre la distribution des distances entre visages voisins pour chaque combinaison de poses. Par exemple, pour la configuration (C,R), nous construisons à partir de l'ensemble d'entraînement un ensemble de paires de visages (C,R), voisins au sens de la CS dans l'espace optimisé pour les paires (C,R). Nous sommes ainsi capables d'estimer :

$$P(Y(q_P)|CS(r_P, q_P)),\ p \in (C, L, R) \tag{4.14}$$

A ce stade, il est désormais possible de trancher : l'identité retournée par le classificateur est alors l'identité associée à la prédiction de probabilité maximale.

Une telle procédure permet de réaliser un gain d'environs 1% sur la précision du classificateur dans le scénario où aucun seuil n'est utilisé pour refuser une prédiction. En faisant usage d'un seuil sur les $P(CS(r_P, q_P))$, on peut faire varier la précision et le rappel obtenus et construire une courbe précision rappel pour le problème de classification décrit plus haut. La figure 4.6 présente la courbe ROC obtenue pour la classification de 1680 individus sur la base de données LFW, avec et sans apprentissage spécialisé par pose. On constate un écart important entre la courbe bleue (sans prise en compte de la pose), et la courbe rouge

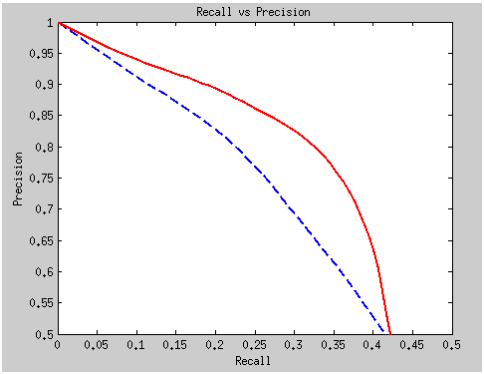

Figure 4.6 Courbe ROC pour la classification de 1680 sujets (avec prise en compte de la pose : rouge, sans pose : bleu)

(avec prise en compte explicite de la pose). Nous pouvons donc conclure que la distinction de l'apprentissage par pose est bénéfique à la classification.

4.2.2 Données d'apprentissage

Nous avons souligné plus haut l'impact du nombre d'exemplaires disponibles pour l'apprentissage sur la précision de la classification. Une question qui se pose naturellement à ce stade est alors : pourquoi ne pas se procurer des exemplaires supplémentaires d'apprentissage ?

Ou ? Se procurer des portraits d'individus annotés n'est pas une tâche triviale. Dès que l'on adresse la grande échelle, la disponibilité de portraits de qualité chute. Prenons par exemple Agbani Darego. Top model célèbre, le web regorge d'images de la jeune femme. En revanche, Alexandra Rozovskaya (musicienne russe) est très mal représentée en ligne et il est difficile d'obtenir un portait d'Alexandra différent de celui de LFW.

L'exemple choisi ici est sans doute un peu extrême. Pour la plupart des individus de LFW ne possédant qu'une image dans la base de données, il est possible de trouver au moins quatre images supplémentaires via une recherche sur Google. Toutefois, un filtrage des visages obtenu est incontournable si l'on souhaite preserver la fiabilité des données d'apprentissage.

T. Berg et al. abordent le problème de la vérification de données collectées automatiquement sur le web (Berg *et al.*, 2004). Leur base de données (Faces in the Wild) de visages est constituée de plus de quarante-quatre mille visages détectés sur plus d'un demi million d'images extraites d'articles d'actualité en ligne. Leur travail consiste à exploiter le contenu (texte) du descriptif de l'image pour identifier le contenu des photographies. Par exemple, il

Figure 4.7 Disponibilité d'images supplémentaires. En haut : Agbani Darego, en bas : Alexandra Rozovskaya

Figure 4.8 La nécessité de procéder à un filtrage

s'agit, lorsque plusieurs visages sont présents, d'assigner une identité à chaque visage. Leur méthode est basée sur une procédure complexe de clustering visant à constituer des groupements d'images cohérents en terme d'aspect visuel, mais aussi d'étiquette associée. En fin de compte, ils obtiennent un étiquetage dont la précision est estimée à 77% (Gionis *et al.*, 1999).

Ainsi, la recherche automatisée d'exemplaires additionnels pour l'apprentissage est loin d'être évidente. Les bases de données sont contraintes de faire un compromis entre le nombre d'instances disponibles et la précision de l'étiquetage. LFW est constituée d'un sous-ensemble

de FW, tel que l'étiquetage est plus beaucoup plus fiable. En choisissant cette base de données, nous prenons le parti de la précision de l'étiquetage. Ajouter à LFW des images externes reviendrait à défaire le travail d'épuration de FW réalisé par Huan et al. Ici, nous n'effectuons pas cette expérience. Il serait tout de même intéressant de tester la précision d'un classificateur sur des données plus nombreuses et plus bruitées de FW, en comparaison aux performances obtenues sur LFW. Ce type d'expérience dépasse malheureusement la cadre de notre travail.

4.2.3 Discussion et conclusion partielle

La pose et le nombre d'exemplaires d'apprentissage sont deux facteurs impactant fortement la précision d'un classificateur de visages. Une prise en compte explicite de la pose permet d'améliorer la précision non seulement en vérification, mais aussi en classification. En ce qui concerne le nombre d'instances d'apprentissage, il semble en revanche qu'on doive effectuer un compromis entre la précision de l'étiquetage et le nombre de visages disponibles par individu. Ainsi, sans algorithme fort de vérification, acquérir de nouveaux visages de manière automatisée est susceptible d'introduire du bruit dans la base de données d'apprentissage, ce qui provoquerait sans doute une perte de précision.

Obj2.2.i : Proposer une stratégie de classification tenant compte explicitement de la pose des individus.

En nous appuyant sur le travail de notre collègue K. Hasan, nous avons proposé une stratégie de classification permettant de distinguer les cas selon la pose des individus. Nous avons montré qu'en utilisant une estimation de la distribution des distances entre paires voisines, nous pouvions améliorer significativement la précision et le rappel obtenus.

Obj2.2.ii : Discuter de la collecte automatisée d'exemplaires supplémentaires pour l'apprentissage.

En ce qui concerne la collecte d'instance supplémentaires pour l'apprentissage, il semble qu'à l'échelle d'LFW, un système de vérification robuste soit nécessaire, ce qui dépasse le cadre de notre mémoire.

4.3 Jeux d'échelles

4.3.1 Position du problème

Pour une vidéo de concert de 15min, le nombre de visages que l'on peut extraire au moyen du détecteur de Viola et Jones est généralement de l'ordre du millier à la dizaine de milliers. Quant au nombre d'identités potentielles pour ces visages, elle dépend de la stratégie choisie. Dans l'idéal, on traitera de la même manière tous les visages trouvés, indépendamment du groupe ou du genre musical. Sachant que sur Wikipédia, on recense plus de 5,000 groupes

de rock and roll, que chaque groupe possède en moyenne quatre à cinq membres, on peut sans prendre trop de risques estimer qu'un système de reconnaissance automatisé de musicien inter genres dépasserait en ampleur la base de données LFW et ses 5749 personnalités. Cette réflexion soulève alors la question de la classification de visages à très grande échelle.

En analyse de contenu vidéo sur le web, l'efficacité de l'étape de classification est un paramètre crucial. Comme nous l'avons souligné au chapitre 2.3, il existe de nombreuses structures de données pour la recherche rapide du plus proche voisin à grande échelle. Bien qu'elles dépassent la recherche brute en terme d'efficacité théorique, il est important de s'interroger quant à la légitimité de leur utilisation. En C++, la recherche par force brute du plus proche voisin de 4300 vecteurs parmi 4800 vecteurs de 200 dimensions prend 21 secondes, soit environs 4ms par requête. à partir de quelle échelle est-il nécessaire d'avoir recours à une structure de données pour la recherche rapide du plus proche voisin ? Quelles structures sont adaptées au problème de classification de visages à grande échelle ? Quels sont les paramètres qui conditionnent l'efficacité de ces structures ?

4.3.2 Structures de données pour la très grande échelle

Nous avons présenté au chapitre 2.3 plusieurs structures de données permettant la recherche rapide du plus proche voisin d'un vecteur x dans un espace métrique X doté de la distance D. notamment, nous avons mis en lumière deux structures de données - l'arbre de métriques et l'arbre à couverture, plus adaptées que le kd-arbre à la recherche rapide du plus proche voisin dans un espace de grande dimension ($d > 10$).

M-arbre. L'arbre de métriques (*metric tree*) est basé sur le même principe que le kd-arbre. Il vise à porter cette structure vers les espaces de dimension plus vaste, en ayant recours à un mécanisme de division de l'espace moins sensible à la dimension des vecteurs. Notamment, il vise à minimiser le nombre de calculs de distance pratiqués au moment de la recherche. L'arbre que nous utilisons ici est une variante du vp-arbre (Kunze et Weske (2010)). Comme le vp-arbre, il s'agit d'un arbre binaire. Dans un vp-arbre, chaque noeud est divisé en fonction de la distance des éléments du noeud à un point de référence (*vantage point*). L'inconvénient des vp-arbres est que rien ne garantit l'équilibre de la structure. Le M-arbre (Houten *et al.*, 2005), lui, sépare les objets contenus dans un noeud en deux enfants de taille fixe, ce qui garantit un certain équilibre.

Au moment de la recherche du plus proche voisin d'une requête, le recours aux distances précalculées lors de la construction de l'arbre permet d'accélérer la procédure : dans chaque noeud, la requête est seulement comparée aux centres de chaque enfant du noeud. L'utilisation de l'inégalité triangulaire permet alors d'éliminer certaines branches de la recherche. Pour une recherche par rayon, c'est à dire de tous les éléments situés autour de q à un distance

inférieure à $r(q)$, l'application de ce principe est très simple. Supposons que l'on se trouve dans le noeud R, possédant deux enfants $R1, R2$. $r(R1)$ est le rayon de la sphère centrée en R1 de rayon $r(R1)$, avec $r(R1)$ la distance de R1 à son enfant le plus éloigné. Alors, si $d(q, R1) > r(q) + r(R1)$, $R1$ et ses descendants peuvent être écartés de la recherche. La recherche des k-plus proche voisin est basée sur le même principe, modulo l'utilisation d'une pile de recherche.

Arbres à couverture. Un arbre à couverture est basé sur la notion de couverture hiérarchique. Nous donnons ici la description de la structure telle que décrite dans (Motwani, 1998). étant donné un jeu de données S, chaque noeud de l'arbre est associé à un élément de S, tandis que chaque élément de S est associé à un ou plusieurs noeuds. Soit z la profondeur de l'arbre T, $i = z - j$ avec j la longueur du chemin de la racine jusqu'au niveau i, C_i l'ensemble des points de S associés au noeuds du niveau i, et s un réel positif. L'arbre à couverture respecte les règles suivantes :

1. $C_i \subset C_{i-1}$.
2. $\forall p \in C_{i-1}$, $\exists q \in C_i$ tel que $d(p, q) < s^i$ et le noeud à la profondeur i associé à q est un parent du noeud associé à p à la profondeur i-1.
3. $\forall (p, q) \in C_i$, $d(p, q) > s^i$.

La recherche du plus proche voisin dans une telle structure est effectuée comme suit :

1. $Q_\infty = C_\infty$, la racine de T.
2. pour i de ∞ à $-\infty$: $Q = \{children(q) | q \in Q_i\}$, $Q_{i-1} = \{q \in Q | d(p, q) < d(p, q) + s^i\}$
3. retourner $min_q(d(p, q) | q \in Q_{-\infty})$

Une telle structure est basée sur la constante d'expansion (Karger et Ruhl, 2002) de l'espace de recherche (terme s^i). La constante d'expansion d'un espace métrique (X,d) est définie comme la plus petite valeur $c > 1$ telle que :

$$\forall p \in X, \forall r \geq 0, |B_X(p, 2r)| \leq c|B_X(p, r)| \tag{4.15}$$

Lorsque la distribution de X est uniforme sur m dimensions, $c = 2^m$. L'arbre à couverture est donc adapté aux espaces métriques dotés d'une structure sous-jacente, telle que les données sont groupées en amas à plusieurs échelles. Il n'est pas adapté aux distributions uniformes (Figure 4.9).

La complexité théorique des étapes de construction, recherche exacte et approchée pour l'arbre à couverture a été établie par (Motwani, 1998). En ce qui concerne le M-arbre, on pourra consulter (Andoni et Indyk, 2008). Nous rapportons ces valeurs dans le tableau 4.3.

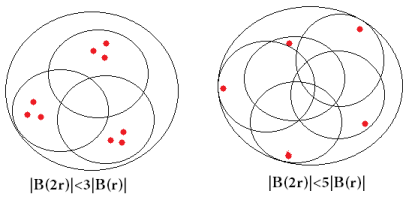

Figure 4.9 Espace métrique et constante d'expansion. à gauche, un espace métrique adapté à l'arbre à couverture ($c = 3$), à droite, un espace métrique mal adapté ($c = 5$)

Tableau 4.3 Complexité des opérations élémentaires pour l'arbre à couverture. Δ est le ratio d'aspect, c'est-à-dire le rapport de la distance maximale sur |X| par la distance minimale sur |X|

	construction	recherche exacte	recherche approchée
Arbre à couverture	$O(c^{16}nlog(n))$	$O(c^{12}log(n))$	$O(log(\Delta)) + \frac{1}{\epsilon}^{O(1)}$
M-arbre	$O(nlog(n))$	$O(log(n))$ sous conditions	-

En pratique. Comme nous l'avons souligné plus haut, l'efficacité des structures de données arborescentes dépend de la distribution des données. Une distribution uniforme nuit à ces structures. Aussi, est-il nécessaire de les évaluer sur des espaces métriques appropriés. Ici, nous proposons une évaluation sur une base de données de visages, projetées dans l'espace X par CSML. La métrique utilisée est la norme euclidienne. La base de donnée utilisée ici, WikipédiaDB, a été constituée par K. Hasan à partir des images de personnalités sur Wikipédia. Elle comporte 51378 visages, dont la description après CSML est constituée de vecteurs normés de 200 dimensions.

Tableau 4.4 Recherche exacte du plus proche voisin (200 dimensions, L_2)

Base de données	méthode	dim	|X|	t(ms) construction	t(ms) test (par requête)
WikipédiaDB	force-brute	200	51329	0	0.015
WikipédiaDB	c-arbre	200	51329	407	0.023
WikipédiaDB	m-arbre	200	51329	209	0.042

Dans un premier temps, nous comparons (Tableau 4.4) les performances des deux structures arborescentes à la recherche par force brute. L'echec est cuisant. En fait, comme l'expliquent Liu et al. (Liu *et al.*, 2004a), ce type de structure de données peine à traiter les vecteurs de grande dimension $d \geq 100$. En terme de recherche approchée, ou dans le cas où plus d'un voisin est requis, les structures hiérarchiques permettent tout de même d'accélé-

rer la procédure de recherche linéaire. Mais la question est alors, dans un contexte où les instances d'apprentissage sont rares, peut-on se permettre de pratiquer la classification par un voisin approché ? Cette question intéressante dépasse malheureusement le cadre de notre travail.

Ici, nous souhaitons retrouver le plus proche voisin exact dans un temps minimal. Face à l'obstacle de la dimensionnalité de la représentation, deux stratégies s'offrent à nous. La première consiste à avoir recours à des techniques de hachage, qui, comme le soulignent Liu et al., conduisent à une perte importante de précision (Andoni et Indyk, 2008). L'alternative au hachage consiste à procéder à la réduction de la dimensionnalité des données. Liu et al. proposent d'avoir recours à la technique de projection randomisée des données d'un espace métrique X vers un espace de dimension inférieure.

Contrairement à ce que laisse entendre l'intuition, la projection randomisée ne s'accompagne pas nécessairement d'une perte de précision majeure. L'idée consiste à dire que le nombre de projections augmentant, la probabilité cumulée obtenue pour le vrai plus proche voisin dépasse la probabilité des autres voisins. Le lemme de Johnson-Lindenstrauss (Johnson et Lindenstrauss, 1984) garantit la préservation de l'information si les matrices de projection sont bien construites.

De manière pratique, obtenir une fonction (matrice) de projection randomisée vérifiant les axiomes du théorème de Johnson-Lindenstrauss n'est pas trivial. Toutefois, récemment, Achlioptas a mis en évidence une matrice de projection M extrêmement simple satisfaisant ces axiomes :

$$M_{i,j} = \begin{cases} \sqrt{3}, & \text{si } r \leq 1/6 \\ 0, & \text{si } 1/6 < r < 2/6 \\ -\sqrt{3} & \text{si } r \geq 2/6 \end{cases} \tag{4.16}$$

avec r un nombre réel aléatoire [2].

Ainsi, il est possible de réduire considérablement la dimension des données en garantissant une perte de précision limitée. La question est, au prix de combien de projection ? Le tableau 4.5 apporte une réponse à cette question. Pour la base de données LFW, on observe une convergence de l'erreur après 350 projections et après 400 pour la base de données LFW+Wiki. Ainsi, pour que la réduction de dimensionnalité soit utile, il est nécessaire d'observer un facteur d'accélération largement supérieur à 400 par passage de 200 à 20 dimensions.

2. En C++, la librairie boost procure des générateurs de nombre aléatoires de qualité acceptable.

Tableau 4.5 Stratégie de réduction de la dimensionnalité

Base de données	dim	dim red	num proj.	acc
LFW	200	200	0	0.32
LFW	200	20	150	0.11
LFW	200	20	300	0.29
LFW	200	20	350	0.30
LFW+Wiki	200	200	0	0.27
LFW+Wiki	200	20	350	0.21
LFW+Wiki	200	20	400	0.26

4.3.3 La question de la métrique

Comme nous avons pu le constater lors de l'examen des structures de données adaptées à la grande échelle, la métrique joue un rôle central pour les opérations de recherche rapide dans une structure arborescente. Or, si l'on revient au problème de classification de visages, la CS n'est pas une métrique, mais une similarité. En particulier, elle ne vérifie pas l'inégalité triangulaire. Un exemple simple suffit à le montrer :

$$CS(x, y) = \frac{xy'}{\|x\|\|y\|} \tag{4.17}$$

$$x = (1, 0), z = (0, 1), y = (1/\sqrt{2}, 1/\sqrt{2}) \tag{4.18}$$

alors :

$$d(x, y) + d(y, z) = 2 - 1,58 < d(x, z) = 1 \tag{4.19}$$

Dans l'évaluation précédente en 4.2.1, nous avons utilisé la distance euclidienne. Or, il n'est pas évident que cette distance soit valide pour travailler sur un espace optimisé pour la CS.

De manière générale, $max(CS(v1, x \in X)) \neq min(L_2(v1, x \in X))$.

En effet, il suffit de considérer les vecteurs :

$$v1 = (1, 0), v2 = (0, 1), v3 = (\sqrt{2}, \sqrt{2}) \tag{4.20}$$

pour constater que rechercher les plus proche voisin au sens de la CS est différent de rechercher le plus proche voisin au sens de L_2. Si v1 est la requête et $X = (v1, v2)$ notre

espace de recherche, on a :

$$L_2(v1, v2) = \sqrt{(v1 - v2)(v1 - v2)'} = \sqrt{(2)} = 1.41 \tag{4.21}$$

$$L_2(v1, v3) = \sqrt{(v1 - v3)(v1 - v3)'} = \sqrt{2 + (1 - \sqrt{2})^2} = 1.47 \tag{4.22}$$

Par ailleurs :

$$CS(v1, v2) = v1 * v2' = 0 \tag{4.23}$$

$$CS(v1, v3) = v1 * v3' = 1.41 \tag{4.24}$$

Ainsi, sur cet exemple, $max(CS(v1, x \in X)) \neq min(L_2(v1, x \in X))$. Par suite, on peut affirmer la propriété 1. Lorsque l'on aborde le cas des vecteurs normés, on peut en revanche montrer l'égalité.

Sur la sphère unité, $max(CS(v1, x \in X)) = min(L_2(v1, x \in X))$.
En effet :

$$\forall (x, y) \in B(0, 1)^2,\ L_2(x, y) = \sqrt{(x - y)(x - y)'} = \sqrt{2(1 - xy')} \tag{4.25}$$

Or, $g(u) = \sqrt{2(1 - u)}$ est monotone décroissante sur $[0, 1]$. D'après les théorèmes sur les composées de fonctions, $g \circ CS$ est donc croissante sur les intervalles de [0,1] où CS est monotone décroissante, et $g \circ CS$ est décroissante sur les intervalles de [0,1] où CS est monotone croissante. En d'autres termes, g adopte les variations inverses de la CS. Par suite, $max(CS(v1, x \in X)) = min(L_2(v1, x \in X))$ sur la boule unité.

Ainsi, d'après la propriété 2, il est légitime de pratiquer la classification par recherche du plus proche voisin sous L_2 dans un espace **normé** optimisé pour la CS. Dans le cas inverse, utiliser L_2 en lieu et place de la CS n'est pas légitime.

4.3.4 Discussion et conclusion partielle

Analyser systématiquement le contexte d'une vidéo pour déterminer quels individus sont susceptibles de s'y trouver n'est pas une démarche viable à l'échelle du web. Un système de reconnaissance de visage orienté pour la reconnaissance de personne rescencera, dans l'idéal, un nombre important d'individus. Dans ce chapitre, nous avons essayé de mesurer l'ampleur d'un tel système, et de discuter des stratégies de classification de visages adaptées à ce type de scénario.

Obj2.3.i : étudier les stratégies de classification de visage à très grande échelle.

Comme le souligne Lowe dans (Muja et Lowe, 2009), la recherche rapide du plus proche voisin dans un espace de grand cardinal et de très grande dimension est un problème difficile. La plupart des structures de données conçues pour les espaces de grand cardinal supportent mal des vecteurs de dimension très élevée. Si la recherche du plus proche voisin approchée est tolérable, on pourra avoir recours à l'arbre à couverture, ou, pour la recherche rapide de k-voisins, à un M-arbre. Pour des dimensions très élevées ($d >> 100$), il faudra avoir recours à une autre catégorie de méthodes, basées sur les techniques de hachage rapide. Pour la recherche rapide du plus proche voisin exact dans un espace de $d = 200$ dimensions (dimension d'un portait après encodage par LBP + CSML), c'est la recherche par force brute qui reste la méthode la plus efficace.

4.4 Conclusion du chapitre

La classification de portraits d'individus non conditionnés est un problème ouvert de la recherche en traitement d'images. Les défis à relever pour rendre les systèmes existants robustes et adaptés au contenu des vidéos du web sont différents de ceux rencontrés par les objets. Notamment, le caractère limité des ressources pour l'apprentissage disponible pour chaque individu impose de fortes contraintes quant au type de classificateur utilisé.

La tâche de vérification visage à visage obtient aujourd'hui d'excellents résultats, même dans le scénario de portraits réalisé sans contraintes sur la pose. Cependant, les résultats de classification d'individus sont très infé des attentes des acteurs économiques du web. Dans ce chapitre, nous avons identifié plusieurs facteurs pénalisant les performances des algorithmes de reconnaissance de visage, en particulier la pose et la limitation des instances d'apprentissage.

Ensuite, nous avons proposé une méthodologie de classification prenant explicitement en compte la pose des individus. Nous avons montré qu'une telle démarche permettait d'améliorer la précision d'un algorithme de classification élémentaire, basé sur la recherche du plus proche voisin dans l'espace résultant de l'apprentissage de la similitude en cosinus. Nous avons également discuté du problème des ressources limitées d'apprentissage. À ce sujet, il semble que le recours à plus d'individus doit s'accompagner d'un travail important de desambiguation. Un classificateur basé sur ce type de données doit alors intégrer la notion d'incertitude de l'étiquetage.

En terme de classification, nous avons posé la question de l'échelle et des structures adaptées. Il semble que pour de grandes dimensions, l'usage de structures de données de type arbre à couverture ou M-arbre ne soit légitime que dans le cas où l'on souhaite procéder à une recherche approximée du plus proche voisin, ou bien à la recherche de k-voisins. Même

dans ce cas de figure, les structures de données restent sensibles à la grande dimension des vecteurs représentant les visages. Pour un traitement effice à très grande échelle, on aura recours à une autre catégorie de techniques, basées sur les fonctions probabilistes de hachage. Le recours à de telles méthodes s'assortit bien sûr d'une hausse de l'imprécision. Tant que celà est possible, on aura donc recours à la recherche brute pour le plus proche voisin exact, ou aux structures de données hiérarchiques pour une recherche approchée limitée à une erreur en ϵ.

Le pas est donc encore grand à franchir en termes de reconnaissance de visages pour disposer d'un système opérationnel pour le traitement de vidéos en ligne. Dans le prochain chapitre, nous discutons de la place qu'occuperait un tel système dans un algorithme d'analyse détaillée du contenu de vidéos de concert.

CHAPITRE 5

INDEXATION PAR LE CONTENU DE VIDEOS MUSICALES EN LIGNE

Dans les chapitres précédents, nous avons donné un aperçu des techniques à la pointe en reconnaissance d'objets et d'individus, adaptées au traitement de contenu vidéo. Ici, nous nous adoptons un point de vue plus applicatif pour traiter un problème concret de l'industrie du web. Les vidéos de concert constituent une part importante du contenu musical en ligne et sont particulièrement difficiles à indexer du fait de leur qualité très variable en dépit des indices textuels.

Par l'analyse du retour des usagers, il est possible, à terme, d'identifier des vidéos de bonne qualité. Cependant, ce processus converge lentement et s'avère, nous le verrons, d'une fiabilité discutable. De plus, il ne garantit pas que toutes les vidéos de qualité seront identifiées. Par ailleurs, si l'analyse du son en complément des métadonnées textuelles apporte un indice important concernant la qualité du contenu, il n'est pas difficile de concevoir les lacunes d'un système d'indexation limité au texte et à l'audio (amalgame sur le groupe, mauvaise qualité de la prise de vue, vidéo hors contexte). Ici, nous proposons d'analyser l'apport d'un système d'analyse de l'image pour l'indexation de vidéos de concert.

5.1 Stratégies d'indexation

5.1.1 Indexation de vidéos

L'indexation de vidéos est un thème vaste, englobant des savoirs faire aussi divers que la segmentation de vidéo, l'analyse de trame sonore, la reconnaissance de parole, la reconnaissance d'objets et de visages, l'analyse et la compréhension de texte.

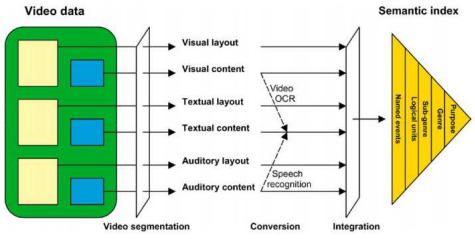

Figure 5.1 Paramètres entrants en jeu dans l'analyse de document vidéo (Snoek et Worring, 2005)

En 2005, Snoek et Worring (Snoek et Worring, 2005) dressent un panorama général des techniques d'analyse de vidéos. Dans une perspective unificatrice, ils proposent une méthodologie unifiée pour l'analyse de documents multimédias basée sur la prise en compte conjointe de l'audio, du texte et de l'image. En terme de données visuelles, ils soulignent deux niveaux d'analyse : le contenu (sujet, ce qui est filmé) et la réalisation (comment le sujet est filmé). Ils distinguent alors trois types de contenus : personnes, objets, et scènes. Les chapitres 3 et 4 de ce mémoire couvrent deux de ces aspects, identifiés par Snoek et Worring comme incontournables pour l'indexation de vidéos.

En 2010, dans une perspective plus pragmatique, Snoek et Smeulders (MédiaMill [1]) décrivent les bonnes pratiques en analyse de contenu vidéo à l'échelle du web (Snoek et Smeulders, 2011). Il détaillent le processus complet d'analyse du contenu, de l'extraction d'information visuelle à la fusion des concepts à haut niveau pour la prise de décision finale. En terme d'analyse d'images, ils soulèvent notamment la problématique des instances étiquetées disponibles pour l'apprentissage, que nous avons traitée au chapitre 3. Plus loin, à propos de la détection d'objets, ils soulignent l'importance de la flexibilité et de l'aptitude à généraliser des détecteurs d'objets (forets de décision versus svm *1-vs-all*), question que nous avons traité au chapitre au même chapitre.

Comment reconnaître une vidéo de concert de qualité ? En 2007 (Snoek *et al.*, 2007), Snoek et al se penchent sur le problème particulier de l'indexation de vidéos de concert. Ils se basent sur une analyse du contenu visuel selon les deux points de vue évoqués dans l'article de 2005(Snoek et Worring, 2005) : les concepts et le style. Ils soulignent notamment que dans une vidéo de concert, à l'inverse d'une vidéo de nouvelles ou d'un site-com, le nombre de concepts d'intérêt est borné, ce qui facilite l'analyse du contenu. Par ailleurs, ils proposent une analyse du style basée essentiellement sur la distance de la caméra à la scène, évaluée à partir de la taille des visages détectés. Ici, nous proposons d'étudier nous aussi les points de vue du style et des concepts, adjoignant à l'analyse des objets d'intérêt une technique d'identification des visages, et à l'étude du style une analyse basée sur l'estimation rapide du flot optique moyen. De plus, nous nous proposons d'intégrer à l'analyse une composante basée sur l'information textuelle et la nature de l'évènement représenté.

5.1.2 Vidéo de concert

Les concepts. Dans (Snoek *et al.*, 2007), les auteurs choisissent 12 concepts d'intérêt basés sur les retours de producteurs de concerts. Les concepts retenus sont : audience (audience), groupe (band), batteur (drummer), visage (face), guitare (guitarist), instruments (instruments), clavier (keybord), personne (person), vue de dos (rear view), scène (stage),

1. http ://www.science.uva.nl/research/mediamill/

et table de mixage (turntable). Ici, nous préférons penser en termes de champs lexicaux, et nous concentrer sur un panel d'instruments diversifié, de manière à traiter une plus grande variété de styles musicaux. Ainsi, nous choisissons les instruments les plus fréquents dans le synset image-net *instruments de musique* : pianos, guitares, batteries, harpes, trompettes, saxophones et accordéons. Nous procédons également à la détection des visages. Au chapitre 3 de ce mémoire, nous nous sommes efforcés de développer un détecteur d'instruments robuste, capable de supporter le contenu, qualifié de difficile par Snoek (Snoek *et al.*, 2007), des vidéos de concert. Au chapitre 4, nous avons proposé une méthodologie pour la détection des visages et la reconnaissance d'individus. Nous les mettons en pratique dans ce chapitre.

Le style. La réalisation d'une vidéo est un autre paramètre important pour juger de la qualité d'une vidéo de concert. Dans (Snoek *et al.*, 2007), Snoek propose d'utiliser la taille des visages détectés pour estimer la distance de la caméra à la scène. Ici, nous proposons de prendre en compte ce paramètre, assorti d'une analyse des mouvements de caméra par analyse du flot optique. Ainsi, nous détectons les mouvements maladroits d'une personne située près de la scène.

Nous avons mentionné plus haut deux points de vue pour l'analyse du contenu de vidéos de concert. Ici, nous traitons du style, ou de la manière dont est filmée la vidéo. Comme l'a mentionné Snoek, on peut utiliser la taille des visages détectés pour estimer la position moyenne de la caméra par rapport à la scène. Nous proposons d'ajouter à cet examen une étude des variations du flot optique moyen par image, illustratif des mouvements de caméra.

Le flot optique correspond au vecteur déplacement de points caractéristiques d'une image. Le flot optique moyen est donc le vecteur déplacement moyen des points clés d'une image. Empiriquement [2], nous avons vérifié que ce vecteur permet de suivre les principaux mouvements de caméra. Afin d'établir la qualité d'une vidéo, nous proposons donc d'étudier le vecteur accélération du flot optique moyen.

$$F_m = \frac{df_m(t)}{dt^2}, f_m(t) = \frac{\sum_{i=1:N}(x(t) - x(t-1))}{N} \tag{5.1}$$

La nature de l'évènement. Pour la nature de l'évènement, nous utilisons les métadonnées textuelles associées à la vidéo, dont nous extrayons des entités nommées bien choisies (nom du groupe, lieu, année). A partir de ces données, nous nous proposons d'évaluer la popularité de l'évènement associé à la vidéo.

Un évènement musical est caractérisé par un groupe, un lieu, une date. Selon les vidéos, une ou plusieurs de ces données sont disponibles dans le titre du document. Dans l'idéal, nous aimerions disposer d'une base de donnée capable d'évaluer la popularité des notions suivantes : (groupe), (groupe+lieu), (groupe+date), (groupe+lieu+date).

2. voir vidéo demoflow.avi

Aujourd'hui, Wikipédia présente un contenu standardisé, qu'il est facile d'exploiter. Notamment, une liste des groupes musicaux célèbres peut être facilement obtenue en minant le contenu du site. Ici, nous nous sommes restreint aux groupes de rock et avons collecté 5941 entrées, de A Band of Orcs à ZZ Top. Tous les groupes n'étant pas de même portée, nous nous proposons de procéder à un filtrage et de ne conserver que les groupes possédant au moins deux images sur leur page Wikipédia.

Tableau 5.1 Nombre d'images par page Wikipédia

Nombre d'images	$>= 0$	> 0	> 1	> 2	> 4	> 8	> 10
Nombre de groupes	5941	3915	1648	639	158	32	11

Les onze groupes possédant le plus d'images sont ACDC, Bomb the Bass, The Human League, Fleetwood Mac, Kiss, Marylin Manson, Metallica, MotherHead, Nine Inch Nails, The Rolling Stones et U2. Toutes considérations subjectives exclues, le filtrage proposé ici semble être raisonnable. Aussi, proposons-nous d'aller plus loin et d'évaluer la popularité d'un groupe par le nombre d'images comptées sur sa page Wikipédia.

Fort de la liste de groupe précédemment établi, nous nous sommes alors posé la question de la recherche d'évênement musicaux associés à ces formations. Nous souhaitons en effet, être en mesure d'anticiper les évènements populaires. Afin de recueillir les évènements associés à ces groupes, nous avons procédé à la fouille de la base de données d'un site de tickets de concerts en ligne (songkick). Le contenu normalisé de ce genre de site (base de données transparente) rend aussi la fouille de données automatisée très facile. Nous avons ainsi collecté 64970 concerts, caractérisés par un groupe, un lieu et une date. Là encore, nous nous sommes proposé d'évaluer la popularité de chaque évènement. Pour ce faire, nous avons lancé une requête sur Google composée du lieu et de la date de l'événement, assorti du mot concert. Nous avons alors procédé au comptage de l'évènement E, pour les chacune des 100 premières réponses, *rep*, retournées par le moteur de recherche :

$$\text{E} : \textit{rep} \supset \text{band} + \text{year} + \text{month} + \text{place}$$

Notre hypothèse consiste à penser qu'un évènement populaire obtiendra un plus haut taux de E parmi les cent premières réponses retournées. En prenant garde à quelques cas critiques (homonymes ville-groupe), on obtient des résultats raisonnables. Pour les 64,970 concerts examinés, le processus ne prend que quelques heures. À titre indicatif, les cinq premiers concerts obtenus sont : U2 à Chicago en 2009, Metallica à Sofia en 2011, U2 à Dublin en 2010, Phish à Denvers en 2011, et les Red Hot Chilli Peppers à Londres en 2011 (pour des données collectées en 2011).

5.2 Protocole expérimental et outils d'évaluation

Pour dix groupes (ou artistes) de rock et de jazz, nous obtenons les vidéos haute définition apparaissant aux premiers rangs dans l'index des vidéos retournées par le moteur de recherche Google pour les mots clé *groupe+live+HD*. Les groupes retenus sont : ACDC, Diana Krall, George Benson, Kiss, Metallica, Norah Jones, Red Hot Chilli Peppers, Sonny Rollins, U2, Wynton Marsalis. Cet ensemble est choisi de sorte à procurer une variabilité dans les styles musicaux, le type de vidéo, et la popularité de l'artiste (ou groupe). Nous collectons également les informations concernant le nombre de vues, le nombre de retours utilisateur positifs (*j'aime*) et négatifs *(je n'aime pas*). Le taux d'appréciation d'une vidéo est alors donné par :

$$t_a = \frac{j'aime}{j'aime + jen'aimepas} \tag{5.2}$$

La figure 5.2 montre la distribution statistique des vues, *j'aime* et *je n'aime pas*. On observe que la variance du nombre de vues est très importante. Par ailleurs, le nombre de *je n'aime pas* par vidéo est très faible par rapport au nombre total de vues.

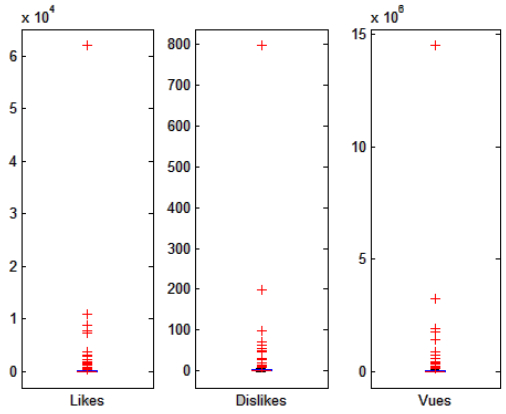

Figure 5.2 Répartition du nombre de vues, j'aime, je n'aime pas pour les dix groupes considérés.

On constate également qu'à nombre égal de vues, les vidéos video3 et video4 [3] sont de qualité très différente. De même, pour U2, la vidéo video6 ne montre pas le groupe U2 mais deux violoncellistes interprétant *With or Without You*, titre populaire ce qui explique le nombre de vues. Les indicateurs découlant des retours des usagers sont donc à prendre avec

3. les vidéos peuvent être visionnées à l'adresse http ://www.youtube.com/watch ?v=CodeDeLaVideo, voir Annexe B

Tableau 5.2 évaluation des vidéos du groupe Red-Hot-Chilli-Peppers et U2 par les retours usagers(extrait)

Vidéo	Groupe	J'aime	Je n'aime pas	Vues	t_a	Quartile (vues)
video1	RedHCP	1763	10	575839	0.9944	4
video2	RedHCP	1704	31	334726	0.9821	4
video3	RedHCP	29	1	8316	0.9667	3
video4	RedHCP	58	1	8479	0.9831	3
video5	U2	74	0	17145	1	3
video6	U2	9021	72	1450974	0.9921	4
video7	U2	118	3	27567	0.9721	4
video8	U2	88	3	29831	0.9670	4

Tableau 5.3 évaluation manuelle des vidéos du groupe Red-Hot-Chilli-Peppers et U2(extrait)

Vidéo	Groupe	Prise de vue	Pertinence	Commentaires
video1	RedHCP	4	4	Pro.
video2	RedHCP	4	4	Pro.
video3	RedHCP	2	1	Amateur loin de la scène.
video4	RedHCP	3	4	Pro.
video5	U2	4	4	Pro
video6	U2	2	1	Hors sujet
video7	U2	4	4	Pro
video8	U2	3	4	Pro. Sombre.

parcimonie. En effet, ces indicateurs ne prennent pas en compte le cas d'une vidéo hors sujet (pas le bon groupe), et différencient mal les vidéos selon la prise de vue. De plus, le nombre de vues d'une vidéo dépend fortement de la popularité de l'auteur, qui peut être un internaute parrainé par un internaute populaire, ou une compagnie commerciale, ainsi que du temps depuis lequel la vidéo est en ligne.

L'évaluation de notre démarche n'est donc pas triviale. Nous proposons donc d'avoir recours à une évaluation humaine de la pertinence de la vidéo et de la qualité de prise de vue en complément du taux d'appréciation t_a. L'échelle d'évaluation est en 4 points : 4 très bon, 3 bon, 2 mauvais, 1 très mauvais, respectivement pour la pertinence (groupe souhaité, concert) et la prise de vue (bon point de vue, prise de vue stable). On défini alors l'évaluation globale d'une vidéo comme :

$$Eval = min(Pertinence, PriseDeVue) \tag{5.3}$$

Le paragraphe suivant concerne la prédiction de cette valeur au moyen des indicateurs présentés jusque ici.

5.3 Indexation de vidéos par le contenu

Nous disposons des indicateurs suivants : groupe musical (G), popularité du groupe (PG), flot-optique moyen (F), longueur de la vidéo (L), nombre de visages détectés (NV), nombre d'instruments détectés (NO). Nous comptons également le nombre de détections du leader du groupe en particulier (NL).

Afin d'illustrer notre étude, nous présentons ici une étude de 8 concerts. La table 5.4 récapitule les résultats obtenus pour quatre concerts du groupe Red-Hot-Chilli-Peppers et de U2.

Tableau 5.4 Données récoltées pour les Red-Hot-Chilli-Peppers et U2 (extrait)

Vidéo	G	PG	F	L	NV	NO	NL	Eval
video1	RedHCPs	0.72	0.24	7980	183	15	102	4
video2	RedHCP	0.72	0.29	11400	230	43	134	4
video3	RedHCP	0.72	0.48	7620	1	23	10	1
video4	RedHCP	0.72	0.33	8220	54	28	25	3
video5	U2	1	0.31	9570	12	57	5	4
video6	U2	1	0.10	6152	77	7	4	1
video7	U2	1	0.31	10350	14	27	5	4
video8	U2	1	0.38	16590	0	27	0	3

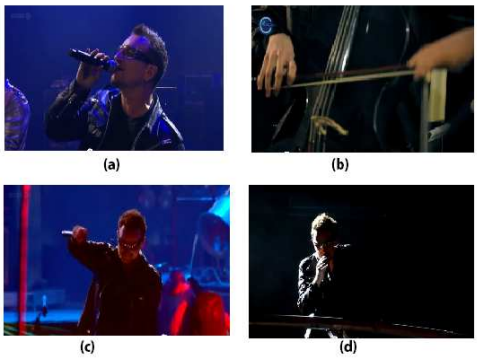

Figure 5.3 Images représentatives des vidéos de U2

Les figures 5.3 et 5.4 donnent un aperçu du contenu des vidéos étudiées à titre d'exemple dans le tableau 5.4. On constate la présence de deux vidéos critiques. La première (Red-Hot, video3, (c)) est filmée depuis la foule. Elle est sombre, on ne voit que rarement la scène et l'auteur de la vidéo bouge fortement lorsque l'audience s'agite. Ceci se traduit par un

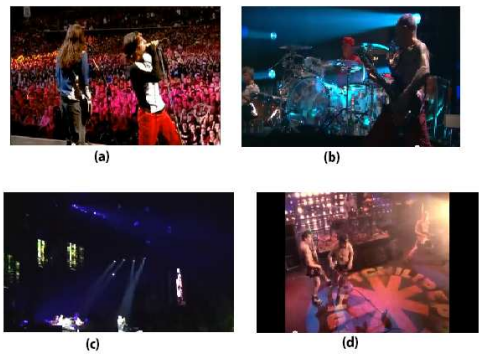

Figure 5.4 Images représentatives des vidéos des Red-Hot-Chilli-Peppers

flot optique (F) plus élevé que la moyenne, un faible nombre de visages et d'instruments détectés (NO,NV). La seconde (U2, video6, (b)) est une vidéo de bonne qualité, mais qui n'est pas une vidéo de U2. L'évaluation de ce type de vidéo est moins aisée. On détecte en effet un flot optique raisonnable, plus bas que la moyenne, car la prise de vue est quasi fixe. Le nombre de visages et d'instruments détectés est important. En revanche, le nombre de visages positivement identifiés comme Bono (NL), est plus faible que la moyenne. Enfin, deux des vidéos sont sombres, ce qui se traduit par un faible nombre de visages et d'objets détectés. Un autre cas critique qui n'est pas illustré ici est celui d'une vidéo statique (ou diaporama amateur), qui se caractérise par un flot optique très bas. Dans ce dernier cas de figure, le nombre d'instruments et de visages est susceptible d'être extrêmement élevé ou nul, suivant le contenu des images statiques.

Prédiction de la prise de vue. Comme le laisse présager notre exemple, l'accélération cumulée du flot optique moyen procure une bonne indication quant à la qualité de la prise de vue. La figure 5.5 indique la répartition de l'évaluation humaine de la prise de vue (ordonnée) par rapport au flot optique. On constate une accumulation des vidéos de bonne qualité autour d'un flot optique à 0.3, et une dégradation de la qualité lorsque le flot optique devient très faible ou très élevé. Cette répartition suggère qu'un séparateur linéaire est insuffisant dans cet espace. Nous avons donc recours à un SVM à noyau quadratique. Le tableau 5.6 indique les résultats obtenus au moyen de ce classificateur appris dans une configuration de *leave-25-out*, c'est à dire en laissant 20% des données de côté à des fin de test, en itérant de sorte à tester l'ensemble des données. La précision moyenne obtenue est de 74%. Si nous répétons à présent l'expérience en ajoutant le nombre d'instruments et de visages détectés. On obtient un gain en précision de l'ordre de 2%.

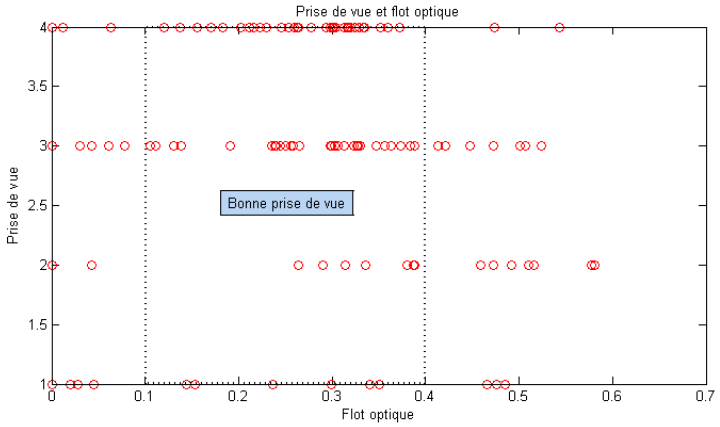

Figure 5.5 Le flot optique permet de prédire la qualité de la prise de vue

Tableau 5.5 Prédiction de la prise de vue à partir du flot optique (SVM (noyau quadratique), leave 25-out)

test	1	2	3	4	5	moyenne
précision	72	65	67	88	81	74

Tableau 5.6 Prédiction de la prise de vue à partir du flot optique (SVM (noyau quadratique), leave 25-out)

test	1	2	3	4	5	moyenne
précision	76	68	66	89	84	76.6

Prédiction de pertinence. En ce qui concerne la prédiction de la pertinence, il s'agit d'identifier, parmi les vidéos de qualité quelles sont les vidéos qui représentent vraiment un membre du groupe désiré. L'étude est plus difficile car on n'a d'autre choix que de reconnaître un ou des membres du groupe. Ici, nous avons opté pour le chanteur principal du groupe. Les résultats sont en pratique assez peu concluants du fait du faible nombre de visages détectés sur les vidéos qui sont propres à l'alignement. Les visages sont en effet souvent de petite taille ou soumis à des jeux d'ombres trop important pour permettre la détection correcte des points clés (nez, coins de yeux, bouche) qui permettent une mise à l'échelle et un alignement sur un axe commun. Pour les vidéos où des visages sont détectés (37), on note une corrélation positive (0.4) entre le rapport du nombre de visages détectés comme le chanteur principal du groupe sur le nombre de visages total et la pertinence de la vidéo. Le faible taux de visages détectés positif s'explique aussi sans doute par le taux d'échantillonnage que nous

avons utilisé (toutes les 10 images). Pour obtenir de bons candidats, sans doute faudrait-il passer en revue toutes les images de la vidéo (multiplier le temps de traitement par 10), ce qui n'est pas raisonnable avec l'implémentation d'openCV du détecteur de Viola et Jones.

Figure 5.6 Limites de l'analyse de visages dans une vidéo de concert

5.3.1 Bilan et conclusion

L'indexation automatisée de vidéos demande la maîtrise de savoirs faire divers, qu'il s'agit non seulement de maîtriser individuellement, mais aussi de combiner intelligemment. L'analyse de contenu vidéo repose sur l'analyse du texte, du contenu visuel, et éventuellement du son. Ici, nous avons proposé une stratégie adaptée au cas particulier des vidéos de concert, basée sur l'estimation de la popularité d'évènements, l'analyse de la qualité de la prise du vue, et la détection de concepts clés et d'individus.

Obj3.1 : Proposer une méthodologie pour la collecte et l'évaluation automatisée de concerts populaires.

Dans ce chapitre, nous avons proposé et mis en oeuvre une méthodologie pour la collecte et l'évaluation d'évènements musicaux de qualité. Nous nous sommes fortement appuyés sur la standardisation des contenus HTML des sites professionnels de grande ampleur, tels que Wikipédia et Songkick. Pour l'évaluation de la popularité d'un évènement, nous avons exploité la trace laissée par un évènement important dans la table d'index du moteur de recherche Google.

Obj3.2 : Proposer une méthode d'indexation et de description du contenu des vidéos en ligne.

Dans un second temps, nous avons proposé et mis en oeuvre une méthodologie pour la construction d'une trace du contenu visuel des vidéos. Nous avons dans un premier temps proposé une méthode capable de détecter les prises de vue de mauvaise qualité. Dans un second temps, nous avons testé les savoirs faire évoqués aux chapitres 3 et 4 pour décrire les concepts et individus contenus dans des vidéos.

Obj3.3 : Proposer une méthode de prédiction de la qualité des vidéos de concert et ligne.

Au terme du cinquième chapitre, nous avons montré que l'étude du flot optique et du nombre d'instruments et de visages détectés permet de prédire la qualité d'une vidéo de concert. Cette analyse donne un outil fiable pour l'identification des contenus de type diaporama et des vidéos amateurs de mauvaise qualité. Cependant, elle n'identifie pas toujours les vidéos de bonne qualité ne représentant pas le groupe souhaité. Pour satisfaire cette dernière attente, il faudrait revoir la procédure d'alignement des visages, relaxer les conditions de détection d'un visage propre à l'alignement, ou éventuellement, traiter non pas le visage, mais le haut du corps comme un objet, doté d'attributs caractéristiques.

CHAPITRE 6

CONCLUSION

L'indexation de vidéos par le contenu est encore un défi pour la recherche en 2012. En effet, nous avons vu qu'il est nécessaire de composer avec les artéfacts d'approximation, d'incertitude, de bruit, et de multinomialité, tant d'obstacles stimulants pour les chercheurs en reconnaissance d'objets et de visages. Dans le cadre de ce mémoire, nous avons étudié en détail les facteurs limitant l'applicabilité des algorithmes de reconnaissance d'objets et d'individus au contenu vidéos du web, et plus spécifiquement, de concert. Nous avons alors proposé des pistes d'amélioration et avant de procéder à une mise en pratique pour l'identification de vidéos mal indexées par le texte les avoisinant.

Ainsi, dans le troisième chapitre, nous avons exploré le domaine de la reconnaissance d'objets et son application aux images complexes. Nous avons alors souligné l'impact des choix en terme de dimensionnalité de la représentation des images et d'encodage sur les possibilités offertes en termes de construction d'un classificateur. Plus spécifiquement, nous avons montré que l'utilisation d'un vocabulaire de moyenne taille permet l'apprentissage d'un classificateur plus complet sans perte de précision majeure. Nous avons également vu que ce choix est supporté par l'analyse des variations de la somme résiduelle des erreurs au moment de la réduction de la dimensionnalité par l'algorithme des k-moyennes. Dans ce même chapitre, nous avons mis en évidence la place centrale des données d'apprentissage pour l'apprentissage d'un classificateur robuste aux images complexes. Constatant l'inapplicabilité à grande échelle d'une procédure d'étiquetage manuel par zone sur de telles images, nous avons proposé une méthode d'étiquetage manuel des images. Enfin, nous avons proposé un classificateur permettant de tirer pleinement profit de cet apprentissage, adapté aux données multinomiales et bruitées à l'image des vidéos de concert en ligne.

Dans le quatrième chapitre, nous avons abordé les défis associés au traitement des visages dans le cadre de l'analyse de contenu vidéo. Nous avons présenté une stratégie originale alliant deux techniques complémentaires d'apprentissage de distance (CSML et LDE) et montré que leur association conduit à un gain de précision non négligeable. Nous avons également étudié deux facteurs critiques qui pénalisent les performances des algorithmes de classification. En particulier, nous avons proposé une stratégie de classification prenant explicitement en compte la pose des individus, dont nous avons montré qu'elle conduit à une amélioration significative des résultats obtenus. Enfin, nous avons discuté de la question de la classification en termes de performances et des structures adaptées pour le passage à des problèmes à très grande

échelle. Nous avons alors rencontré la problématique de la grande dimension de l'espace de recherche lors de l'usage de structures hiérarchiques et exploré plusieurs pistes de solution, en particulier la réduction de dimensionnalité par projection aléatoire.

Dans le cinquième chapitre, nous avons adopté un point de vue plus appliqué pour adresser le problème de l'indexation de vidéos de concert. Nous avons constaté que la combinaison des informations concernant la prise de vue, la présence d'objets clés, et la reconnaissance de visages sont des outils indispensables pour une bonne indexation. Par ailleurs, nous avons constaté que notre système de mesure du flot optique et de détection d'instruments permet d'identifier des vidéos parasites qui ne sont pas filtrées par une analyse du texte avoisinant (amateur, hors sujet, diaporama). Pour finir, nous avons également montré que l'identification des visages détectés apporte à l'analyse, bien qu'elle ne soit pas systématiquement praticable du fait des conditions dans lesquelles sont présentés les visages des artistes.

RÉFÉRENCES

ANDONI, A. et INDYK, P. (2008). Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. *Communications of the ACM*. 117–122.

ARTHUR, D. et VASSILVITSKII, S. (2006). How slow is the k-means method ? 144–153.

B. FROHLICH, E. RODNER, M. K. J. D. (2011). Efficient gaussian process classification using random decision forests. *Mathematical Theory of Pattern Recognition - MTPR*, 21(2), 184–187.

BACH, F. et PONCE, J. (2010). Discriminative clustering for image co-segmentation. *Proceedings of the Conference on Computer Vision and Pattern Recognition - CVPR*.

BANGPENG, Y. et LI, F. F. (2010). Grouplet : A structured image representation for recognizing human and object interactions. *Proceedings of the International Conference on Computer Vision and Pattern Recognition - CVPR*. 9–16.

BELHUMEUR, P. N., HESPANHA, J. P. et KRIEGMAN, D. J. (1997). Eigenfaces vs. fisherfaces : Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence - TPAMI*, 19, 711–720.

BERG, T. L., BERG, A. C., EDWARDS, J., MAIRE, M., WHITE, R., TEH, Y. W., LEARNED-MILLER, E. et FORSYTH, D. A. (2004). Names and faces in the news. *Proceedings of the International Conference on Computer Vision and Pattern Recognition - CVPR*.

BEYGELZIMER, A., KAKADE, S. et LANGFORD, J. (2006). Cover trees for nearest neighbor. *Proceedings of the International Conference on Machine Learning - ICML*. 97–104.

BEZDEK, J. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*.

BOIMAN, O., SHECHTMAN, E. et IRANI, M. (2008). In defense of nearest-neighbor based image classification. *Proceedings of the Conference on Computer Vision and Pattern Recognition - CVPR*. 1–8.

BOTTOU, L. et BENGIO, Y. (1995). Convergence properties of the k-means algorithms. *Proceedings of Advances in Neural Information Processing Systems - NIPS*. MIT Press, 585–592.

BOUACHIR, W., KARDOUCHI, M. et BELACEL, N. (2009). Improving bag of visual words image retrieval : A fuzzy weighting scheme for efficient indexation.

CHEN, H., CHANG, H. et LIU, T. (2005). Local discriminant embedding and its variants. vol. 2, 846–853.

CHEN, Q., SONG, Z., LIU, S., CHEN, X., YUAN, X., CHUA, T., YAN, S., HUA, Y., HUANG, Z. et SHEN, S. (2010). Boosting classification with exclusive context.

CLARKSON, K. L. (1997). Nearest neighbor queries in metric spaces. *Proceedings of the ACM Symposium on Theory of Computing*. 609–617.

CRIMINISI, A., SHOTTON, J. et KONUKOGLU, E. (2012). Decision forests : A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Foundations and Trends in Computer Graphics and Vision*, 7, 81–227.

CSURKA, G., DANCE, C. R., FAN, L., WILLAMOWSKI, J. et BRAY, C. (2004). Visual categorization with bags of keypoints. *Workshop on Statistical Learning in Computer Vision - ECCV*. 1–22.

DENG, J., BERG, A. C., LI, K. et LI, F. (2009). Boosting associated pairing comparison features for pedestrian detection. *In ICCV Workshop on Visual Surveillance*.

DOMENICONI, C., PENG, J. et GUNOPULOS, D. (2002). Locally adaptive metric nearest-neighbor classificaton. *IEEE Pattern Analysis and Machine Intelligence*, 24, 1281–1285.

DONG, L., TAO, L. et XU, G. (2010). Head pose estimation using covariance of oriented gradients. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing - CASSP*. 1470–1473.

DRINEAS, P., FRIEZE, A., KANNAN, R., VEMPALA, S. et VINAY, V. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control - TAC*, 19, 716–723.

DRINEAS, P., FRIEZE, A., KANNAN, R., VEMPALA, S. et VINAY, V. (2004). Clustering large graphs via the singular value decomposition. *Machine Learning Journal*, 56, 9–33.

G. SCHINDLER, M. BROWN, R. S. (2007). City-scale location recognition. *Proceedings of the Conference on Computer Vision and Pattern Recognition - CVPR*.

GIONIS, A., INDYK, P. et MOTWANI, R. (1999). Proceedings of the 25th international conference on very large data bases - vldb. 518–529.

GOLDBERGER, J., ROWEIS, S. T., HINTON, G. E. et SALAKHUTDINOV, R. (2004). Neighbourhood components analysis. *Proceedings of the Conference on Neural Information Processing Systems - NIPS*.

GUILLAUMIN, M., VERBEEK, J. et SCHMID, C. (2009). Is that you ? metric learning approaches for face identification. *Proceedings of the International Conference on Computer Vision - ICCV*.

H. FU, Q. ZHANG, G. Q. (2012). Random forest for image annotation. *Proceedings of the 12th European Conference on Computer Vision - ECCV*.

HASAN, M. K., PUECH, F. et PAL, C. J. (2012). Improving face verification in the wild with poses, and scaling up recognition with discriminative metric data structures, unpublished.

HESS, R. (2010). An open source sift library. *Proceedings of the ACM Multimedia - ACMM)*.

HO, H.-H., KUO, B.-C., TAUR, J.-S. et LI, C.-H. (2008). A Flexible Metric Nearest-Neighbor Classification based on the Decision Boundaries of SVM for Hyperspectral Image. *Proceedings of the International Conference on Geoscience and Remote Sensing IEEE International Symposium - CGRS*. 212–215.

HOUTEN, Y. V., NACI, S. U., FREIBURG, B., EGGERMONT, R., SCHUURMAN, S., HOLLANDER, D., REITSMA, J., MARKSLAG, M., KNIEST, J., VEENSTRA, M. et HANJALIC, A. (2005). The multimedian concert-video browser. *Proceedings of the International Conference on Multimedia Computing and Systems/International Conference on Multimedia and Expo*. 1561–1564.

HSU, C. et LIN, C.-J. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transaction on Neural Networks - TNN*, 13, 415–425.

HUANG, G. B., RAMESH, M., BERG, T., LEARNED-MILLER, E. et HANSON, A. (2005). Labeled faces in the wild : A database for studying face recognition in unconstrained environments.

JACOBS, D. et HUYNH, P. (2010). An evaluation of nearest neighbor images to classes versus nearest neighbor images to images.

JIA, Y., HUAN, C. et DARELL, T. (2012). Beyond spatial pyramids : Receptive field learning for pooled image features.

JOHNSON, W. et LINDENSTRAUSS, J. (1984). Extensions of lipschitz maps into a hilbert space. *Proceedings of the Conference in modern analysis and probability*. vol. 26, 189–206.

KANUNGO, T., MOUNT, D. M., NETANYAHU, N. S., PIATKO, C., SILVERMAN, R. et WU, A. Y. (2004). An efficient k-means clustering algorithm : Analysis and implementation. *Computational Geometry : Theory and Application - CGTA*, 28, 89–112.

KARGER, D. R. et RUHL, M. (2002). Finding nearest neighbors in growth-restricted metrics. *Proceedings of the ACM Symposium on Theory of Computing*. 741–750.

KIM, G. et TORRALBA, A. (2006). Unsupervised detection of regions of interest using iterative link analysis. *Proceedings of the Conference on Neural Information Processing Systems - NIPS*.

KIM, G. et XING, E. P. (2012). On multiple foreground cosegmentation. *Proceedings of the Conference on Computer Vision and Pattern Recognition - CVPR*.

KIM, T. et KITTLER, J. (2005a). Locally linear discriminant analysis for multi-modally distributed classes for face recognition with a single model image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27, 318–327.

KIM, T. et KITTLER, J. (2005b). Locally linear discriminant analysis for multi-modally distributed classes for face recognition with a single model image. *Discrete Dynamics in Nature and Society - DISCRETE DYN NAT SOC*, 27, 318–327.

KUNZE, M. et WESKE, M. (2010). Metric trees for efficient similarity search in large process model repositories. *Proceedings of the International Conference on Business Process Management - BPM*. 535–546.

LEE, D. T. et WONG, C. K. (1977). Worst-case analysis for region and partial region searches in multidimensional binary search trees and balanced quad trees. *Acta Informatica*, 9, 23–29.

LIU, T., MOORE, A., GRAY, A. et YANG, K. (2004a). An investigation of practical approximate nearest neighbor algorithms. *Proceedings of the International Conferenec on Neural Information Processing Systems - NIPS*.

LIU, T., MOORE, A., GRAY, A. et YANG, K. (2004b). An investigation of practical approximate nearest neighbor algorithms. *Proceedings of the International Conference on Neural Information Processing Systems - NIPS*.

LIU, T., MOORE, A. W., GRAY, A. et YANG, K. (2004c). An investigation of practical approximate nearest neighbor algorithms.

LOWE, D. G. (1999). Object recognition from local scale-invariant features. 1150–1157.

M. K. HASAN, C. P. (2011). Improving alignment of faces, for recognition. *Proceedings of IEEE Symposium on Robotic and Sensors Environments (ROSE)*. 249–254.

MANEEWONGVATANA, S. et MOUNT, D. M. (1999). It's okay to be skinny, if your friends are fat.

MATOUSEK, J. (2000). On approximate geometric k-clustering. *Discrete and Computational Geometry*, 24, 61–84.

MIKOLAJCZYK, K. (2006). Multiple object class detection with a generative model. *Proceedings of Computer Vision and Pattern Recognition - CVPR*. 26–36.

MOORE, A. W. (1991). An intoductory tutorial on kdtrees.

MOTWANI, P. I. R. (1998). Approximate nearest neighbors : towards removing the curse of dimensionality. 604–613.

MUJA, M. et LOWE, D. (2009). Fast approximate nearest neighbors with automatic algorithm configuration. *Proceedings of the International Conference on Computer Vision Theory and Applications*. 331–340.

NGUYEN, H. V. et BAI, L. (2010). Cosine similarity metric learning for face verification. 709–720.

NISTER, D. et STEWENIUS, H. (2006). Computer vision and pattern recognition,ieee computer society conference on. *INFORMS Journal on Computing*, 2, 2161 – 2168.

NOROUZI, M., RANJBAR, M. et MORI, G. (2009). Stacks of convolutional restricted boltzmann machines for shift-invariant feature learning. *Proceedings of the Conference on Computer Vision and Pattern Recognition - CVPR*. 2735–2742.

OJALA, T., PIETIKAINEN, M. et HARWOOD, D. (1994). Performance evaluation of texture measures with classification based on kullback discrimination of distributions. *Proceedings of the International Conference on Pattern Recognition - ICPR*. vol. 1, 582–585.

PAINHO, M. et BACAO, F. (2000). Using genetic algorithms in clustering problems. *Proceedings on Geometric Computation*.

PERRONNIN, F., AKATA, Z., HARCHAOUI, Z. et SCHMID, C. (2012). Towards good practice in large-scale learning for image. *Proceedings of the Conference on Computer Vision and Pattern Recognition - CVPR*. 3482–3489.

PERRONNIN, F., LIU, Y., SANCHEZ, J. et POIRIER, H. (2010). Large-scale image retrieval with compressed fisher vectors. 3384–3391.

PERRONNIN, F. et SANCHEZ, J. (2011). Compressed fisher vectors for lsvrc. *PASCAL ImageNet workshop ICCV*. 23–29.

PHILBIN, J., ISARD, M., SIVIC, J. et ZISSERMAN, A. (2007). Object retrieval with large vocabularies and fast spatial matching. *Proceedings of Computer Vision and Pattern Recognition - CVPR*.

QUELLEC, G., LAMARD, M., BEKRI, L., CAZUGUEL, G., ROUX, C. et COCHENER, B. (2010). Medical case retrieval from a committee of decision trees. *IEEE Transactions on Information Technology in Biomedicine - TITB*, 14, 1227–1235.

RICCI, E. et ODOBEZ, J. (2009). Learning large margin likelihoods for realtime head pose tracking. *Proceedings of the IEEE International Conference on Image Processing*. 2593–2596.

RIM, D. et PAL, C. J. (2011). Semi supervised learning for wild faces and video.

SELIM, S. Z. et ALSULTAN, K. (1991). A simulated annealing algorithm for the clustering problem. *Pattern Recognition - PR*, 24, 1003–1008.

SNOEK, C. et SMEULDERS, A. (2011). Internet video search. *Proceedings of the ACM Multimedia*.

SNOEK, C. G. M. et WORRING, M. (2005). Multimodal video indexing : A review of the state-of-the-art. *Multimedia Tools and Applications*, 25, 5–35.

SNOEK, C. G. M., WORRING, M., SMEULDERS, A. W. M. et FREIBURG, B. (2007). The role of visual content and style for concert video indexing. *Proceedings of the International Conference on Multimedia Computing and Systems/International Conference on Multimedia and Expo*. 252–255.

STONE, Z., ZICKLER, T. et DARRELL, T. (2010). Toward large-scale face recognition using social network context. *Proceedings of the IEEE*, 98, 1408 – 1415.

TIEU, K. H. et VIOLA, P. (1999). Boosting image database retrieval. *Proceedings of the Conference on Computer Vision and Pattern Recognition - CVPR*. vol. 1, 228–235.

TORRALBA, A., FERGUS, R. et FREEMAN, W. T. (2008a). 80 million tiny images : a large dataset for non-parametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence - PAMI*, 30, 1958–1970.

TORRALBA, A. B., FERGUS, R. et WEISS, Y. (2008b). Small codes and large image databases for recognition. *Proceedings on the Conference on Computer Vision and Pattern Recognition - CVPR*. 1–8.

TSANG, I. W., CHEUNG, P. et KWOK, J. T. (2005). Kernel relevant component analysis for distance metric learning. *Proceedings of the International Symposium on Neural Networks - ISNN*.

TURK, M. et PENTLAND., A. (1991). Face recognition using eigenfaces. *Proceedings of the International Conference on Computer Vision and Pattern Recognition - CVPR*.

VALSTAR, M., MARTINEZ, B., BINEFA, X., PANTIC, M. et PANTIC, M. (2010). Facial point detection using boosted regres-sion and graph models. *Proceedings of the International Conference on Computer Vision and Pattern Recognition - CVPR*. 2729–2736.

VAN GEMERT, J. C., VEENMAN, C. J., SMEULDERS, A. W. M. et GEUSEBROEK, J. M. (2010). Visual word ambiguity. *IEEE Transactions on Pattern Analysis and Machine Intelligence - PAMI*, 32, 1271–1283.

VIOLA, P. et JONES, M. (2001). Robust real-time object detection. *Proceedings of the International Journal of Computer Vision - IJCV*.

VUKADINOVIC, D. et PANTIC, M. (2005). Fully auto-matic facial feature point detection using gabor fea-ture based boosted classifiers. *Proceedings of the International Conference on Systems, Man and Cybernetics - CSMC*. 1692–1698.

WANG, J., YANG, J., YU, K., LV, F., HUANG, T. et GONG, Y. (2010). Locality-constrained linear coding for image classification. *Proceedings of the Conference on Computer Vision and Pattern Recognition - CVPR*. 3360–3367.

WEINBERGER, K., BLITZER, J. et SAUL, L. (2005). Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research - JMLR*, 10.

WOLF, L., HASSNER, T. et MAOZ, I. (2011). Face recognition in unconstrained videos with matched background similarity. *Proceedings of the International Conference on Computer Vision and Pattern Recognition - CVPR*. 529–534.

WOLF, L., HASSNER, T. et TAIGMAN, Y. (2009a). Similarity scores based on background samples. *Proceedings of the Asian Conference on Computer Vision - ACCV*. 88–97.

WOLF, L., HASSNER, T. et TAIGMAN, Y. (2009b). Similarity scores based on background samples. 88–97.

YAKHNENKO, O., VERBEEK, J. et SCHMID, C. (2011). Region-based image classification with a latent svm model.

YAO, B., BRADSKI, G. et FEI-FEI, L. (2011a). A codebook-free and annotation-free approach for fine-grained image categorization.

YAO, B., KHOSLA, A. et FEI-FEI, L. (2011b). Combining randomization and discrimination for fine-grained image categorization.

YIANILOS, P. N. (1993). Data structures and algorithms for nearest neighbor search in general metric spaces. *Proceedings of ACM-SIAM Symposium on Discrete Algorithms*. 311–321.

YIN, Q., TANG, X. et SUN, J. (2011). An associate-predict model for face recognition. *Proceedings of the International Conference on Computer Vision and Pattern Recognition - CVPR*. 497–504.

ZHANG, H., BERG, A. C., MAIRE, M. et MALIK, J. (2006). Svm-knn : Discriminative nearest neighbor classification for visual category recognition. *Proceedings of the Conference on Computer Vision and Pattern Recognition - CVPR*. vol. 2, 2126–2136.

ANNEXE A

Bases de données expérimentales

A.1 Reconnaîssance d'objets

Afin de mesurer empiriquement les performances de nos algorithmes nous utilisons trois jeux de données. Les deux premiers rassemblent plusieurs classes d'instruments musicaux.

Tableau A.1 Propriétés des bases de données utilisées

Nom	Nb. classes	Nb. moyen d'images par classe	Num SIFT
GoogleDB	7	103	250K
ImageNetDB	7	1374	9M
PascalDB	20	417	2M

La base de données GoogleDB regroupe les catégories accordéon, guitare, piano, saxophone, trompette, harpe, et batterie. Elle est constituée d'images provenant d'une requête sur le moteur de recherche Google pour des images sans arrière plan (PNGs). La seconde base de données, ImageNetDB est constituée des mêmes catégories et consiste en un sous-ensemble du synset *instruments de musique* sur image-net. Enfin, la base de données PascalDB correspond à la base de données fournie par le PascalDB VOC de 2007 (toujours d'actualité aujourd'hui).

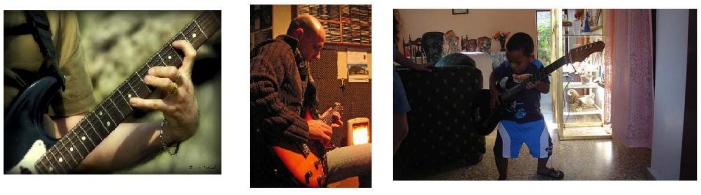

Figure A.1 Extrait de la base de données ImageNetDB

A.2 Reconnaîssance de visages

En ce qui concerne la reconnaîssance de visages, nous utilisons la base de données de référence Labelled Faces in the Wild (LFW).

ANNEXE B

Vidéos, référence des exemples

Les vidéos peuvent être visionnées en ligne : http :-slash-slash-www.youtube.com-slash-watch ?v=CodeDeReference.

Tableau B.1 Correspondance nom des vidéo dans le texte et code de référence sur Youtube

Nom	Code sur Youtube
video1	98LlgYPVVz4
video2	G0rQyIIS15k
video3	gZWljZoD1SA
video4	HeEOW1Ak8zc
video5	42fo3jRWkto
video6	oNtali-cuYA
video7	pYNbbRA3TOI
video8	vqQW-ORqjik