UNIVERSITE SULTAN MOULAY SLIMANE **N° d'ordre : .....**

Faculté des Sciences et Techniques

Béni-Mellal

Centre d'Etudes Doctorales « Sciences et Techniques »

Formation Doctorale « Mathématiques et Physiques Appliqués »

THESE

Présentée par

Mustapha OUJAOURA

Pour obtenir le grade du

Doctorat

Spécialité : **Informatique**

Annotation Automatique d'Images

Soutenue le 12/04/2014 devant la commission d'examen :

Président	**Mr. Said MELLIANI**	**PES**	**FST de Béni-Mellal**
Rapporteurs	**Mr. Ahmed EL KHADIMI**	**PH**	**INPT de Rabat**
	Mr. Said SAFI	**PH**	**FP de Béni-Mellal**
	Mr. Mohamed SABRI	**PH**	**FST de Béni-Mellal**
Examinateurs	**Mr. Cherki DAOUI**	**PH**	**FST de Béni-Mellal**
	Mr. Mohamed FAKIR	**PES**	**FST de Béni-Mellal**
	Mr. Brahim MINAOUI	**PH**	**FST de Béni-Mellal**

2014

DÉDICACES

A la mémoire de mon grand père

A ma famille

A mes amis

A tous ceux qui me sont chers

REMERCIEMENTS

Le travail présenté dans ce document a été effectué au sein de l'Equipe de Traitement de l'Information et de Télécommunications (TIT) du Laboratoire de Traitement de l'Information et Aide à la Décision (TIAD) affilié au Centre d'Etudes Doctorales de la Faculté des Sciences et Techniques (FST) à l'Université Sultan Moulay Slimane de Béni-Mellal.

Je tiens tout d'abord à exprimer ma plus profonde estime et mes vifs remerciements à mon directeur de thèse le professeur Brahim MINAOUI pour sa disponibilité permanente, son soutien persévérant et ses conseils pertinents. Aussi, je remercie sincèrement mon co-directeur de thèse le professeur Mohamed FAKIR pour son encadrement considérable, ses idées précieuses et son soutien constant.

Je remercie vivement tous les membres de jury de ma thèse qui ont pris de leurs temps pour lire et juger mon travail ainsi que pour leur déplacement le jour de la soutenance. Egalement, je remercie infiniment les rapporteurs de ma thèse pour leurs corrections et leurs remarques appropriées vis-à-vis de mon manuscrit.

Mes chaleureux remerciements à mon oncle Bassou HAMRI, mes parents, mes frères, mes sœurs et tous les membres de ma famille qui m'ont aussi apporté leur aide nécessaire au cours de mes études et qui m'ont supporté dans les moments de stress et difficultés.

En fin, un grand hommage à tous mes collègues et amis du Lycée Qualifiant Tarik Ibn Ziad d'El-Ksiba, de la Faculté des Sciences et Techniques de Béni-Mellal et de l'Equipe de Traitement de l'Information et de Télécommunications (TIT).

RÉSUMÉ

La croissance rapide d'Internet et de l'information multimédia a engendré un besoin en technique d'indexation et de recherches d'information multimédia, et plus particulièrement en recherche d'images. Des systèmes de recherche d'images ont été développés pour permettre de faire des recherches dans des bases de données d'images. Cependant ces systèmes sont toujours peu performants en ce qui concerne la recherche sémantique d'images par requête textuelle. Pour effectuer une recherche sémantique, il faut être en mesure de transformer le contenu visuel des images (couleurs, textures, formes) en informations sémantiques. Cette transformation, appelée annotation d'images, assigne une légende ou des mots clés à une image numérique. Les méthodes traditionnelles de recherche d'image reposent fortement sur l'annotation manuelle d'image qui est très subjective, très coûteuse et devient impossible étant donné la taille et la croissance phénoménale des bases de données d'images actuellement existantes. C'est donc tout naturellement qu'a émergé la recherche pour trouver une solution informatique au problème. C'est ainsi qu'ont rapidement fleuri des travaux de recherche sur l'annotation automatique d'images visant à réduire aussi bien le coût d'annotation que le fossé sémantique séparant les concepts sémantiques et les caractéristiques numériques de bas niveau. Notre contribution dans ce domaine est de proposer un système d'annotation automatique d'image plus performant.

Afin d'atteindre cet objectif, nous nous sommes penchés, dans un premier temps, sur l'étude bibliographique des différentes approches d'annotation automatiques d'images. Cette étude nous a permis de concevoir la structure du système d'annotation automatique réalisé. En suite, pour mettre en œuvre ce système, nous avons implémenté plusieurs algorithmes de classification, d'extraction des descripteurs et de segmentation d'images. Ces algorithmes ont été implémentés de façon à être modulaires et ainsi permettre de substituer différentes techniques les unes aux autres afin de choisir la combinaison la mieux adaptée. Pour optimiser le fonctionnement du système, des études expérimentales d'annotation automatique d'images sont réalisées sur trois bases de données d'images. L'analyse des résultats de ces études nous a permis de perfectionner l'architecture fonctionnelle du système d'annotation d'images considéré. L'architecture la plus performante est définie par :

- La combinaison de deux approches de classification différentes et complémentaires à savoir l'approche discriminative et l'approche générative. Les études expérimentales nous ont conduites à combiner le classificateur discriminatif (réseau de neurones) avec le classificateur génératif (réseau bayésien).

- La combinaison des descripteurs. Les tests que nous avons réalisés ont montré que la combinaison des descripteurs est plus performante que leur fusion, et que les descripteurs utilisés doivent décrire la forme, la texture et la couleur des objets à annoter. Ainsi nous avons combiné le descripteur de forme, moments de Legendre, avec celui de la texture, matrice de cooccurrence, et celui de la couleur, histogrammes de couleur RGB.

- Le regroupement des régions adjacentes, résultantes d'une segmentation automatique d'images, pour avoir des objets sémantiquement compacts.

Cette architecture originale permet :

- L'intégration d'autres descripteurs pour améliorer davantage les performances du système d'annotation automatique d'images.

- L'intégration d'autres modules pour décrire l'objet par sa couleur, sa texture et/ou sa taille, … etc. Dans ce travail, nous avons ajouté, au système d'annotations automatiques réalisées, un module qui désigne la couleur dominante d'un objet.

- La prise de décision par le vote des classificateurs combinés. Le mot-clé ayant le maximum de votes est considéré comme étant le mot le plus probable pour l'annotation d'un objet.

Le système d'annotation automatique d'images ainsi réalisé est utilisé pour annoter des images de trois bases d'images à savoir ETH-80, COIL-100 et une base construite au cours de ce travail appelée NATURE. Les résultats obtenus sont très satisfaisants.

Pour passer à l'échelle, c'est à dire à d'autres bases plus larges de données images complexes, il faut améliorer la segmentation, intégrer d'autres descripteurs et changer le langage de programmation.

Notons aussi que nous avons appliqué les algorithmes de classification, d'extraction des descripteurs et de segmentation, implémentés au cours de ce travail, à la reconnaissance des caractères Tifinagh.

Titre : Annotation automatique d'images.

Mots clés: image, segmentation, descripteurs, apprentissage, reconnaissance, classification, réseaux de neurones, réseaux bayésiens, annotation, fossé sémantique.

ABSTRACT

The rapid growth of the Internet and multimedia information has generated a need for technical indexing and searching of multimedia information, especially in image retrieval. Image searching systems have been developed to allow to search in image databases. However these systems are still inefficient in terms of semantic image searching by textual query. To perform semantic searching, it is necessary to be able to transform the visual content of the images (colours, textures, shapes) into semantic information. This transformation, called image annotation, assigns a legend or keywords to a digital image. The traditional methods of image retrieval rely heavily on manual image annotation that is very subjective, very expensive and impossible given the size and a phenomenal growth of currently existing image databases. It is thus quite naturally that the research has emerged in order to find a computing solution to the problem. It is thus that research work were quickly bloomed on the automatic image annotation, aimed at reducing both the cost of annotation and the semantic gap between semantic concepts and digital low-level features. Our contribution in this field is to propose an automatic image annotation system more efficient.

To achieve this goal, we focused, at first, on the literature review of different automatic image annotation approaches. This study has helped and allowed us to design the structure of the achieved automatic image annotation system. Next, to implement this system, we have implemented several algorithms for classification, extraction of descriptors and image segmentation. These algorithms have been implemented so as to be modular and allow different techniques to substitute each other in order to choose the most suitable combination. To optimize the operation of the system, the experimental studies of the automatic image annotation are performed on three image databases. Analysis of the results of these studies allowed us to improve the functional architecture of the considered image annotation system. The most efficient architecture is defined by:

- The combination of two different and complementary classification approaches, namely the discriminative approach and the generative approach. The experimental studies have led us to combine a discriminative classifier (neural network) with a generative classifier (Bayesian network).

- The combination of descriptors: The tests we carried out have shown that the combination of descriptors is more efficient than their fusion, and the used descriptors should describe the shape, texture and colour of objects to annotate. So we combined the shape descriptor, Legendre moments, with the texture, co-occurrence matrix, and the color, RGB colour histograms.

- The grouping of adjacent regions, resulting in automatic images segmentation, in order to have semantically compact objects.

This original architecture allows:

- Integration of other descriptors to further improve the performance of automatic image annotation system.

- Integration of other modules to describe the object by its color, texture and / or size, etc… In this work, we added, to the performed automatic annotation system, a module which determines the dominant color of an object.

- The decision by a vote of combined classifiers. The keyword with the maximum votes is considered to be the most likely word for annotating an object.

The automatic image annotation system thus produced is used to annotate images of three image databases namely ETH-80, COIL-100 and a base built during this work called NATURE. The obtained results were very satisfactory.

To pass to large-scale, i.e. to other larger databases of complex images, we must improve the segmentation, integrate other descriptors and change the programming language.

Also note that we applied the segmentation, extraction of the descriptors, and classification algorithms implemented in this work for the recognition of Tifinagh characters.

Title: Automatic image annotation.

Key-words: image, segmentation, descriptors, learning, recognition, classification, neural networks, Bayesian networks, annotation, semantic gap.

ملخص

لقد ولد النمو السريع للإنترنت و المعلومات المتعددة الوسائط حاجة ملحة لتقنية الفهرسة و البحث عن المعلومات المتعددة الوسائط، وخاصة تقنية البحث عن الصور. وقد تم تطوير نظم البحث في الصور لتسهيل البحث في قواعد بيانات الصور. ولكن هذه الأنظمة لا تزال غير فعالة من حيث البحث الدلالي عن الصور من خلال الاستعلام النصي. لإجراء البحث الدلالي ينبغي أن تكون تلك النظم قادرة على تحويل المحتوى المرئي للصور (الألوان، والقوام والأشكال) إلى معلومات دلالية. هذا التحويل، المسمى بشرح الصور، يخصص تعليق أو كلمات رئيسية لصورة رقمية. تعتمد الطرق التقليدية للبحث عن الصور بشكل كبير على الشرح اليدوي للصور الذي يتميز بكونه ذاتي ومكلف جدا ومستحيل نظرا للحجم والنمو الهائل في قواعد بيانات الصور الموجودة حاليا. لذلك من الطبيعي أن يبرز هناك بحث عن تقنية لحل المشكلة. وقد أدى ذلك إلى ازدهار البحوث بسرعة عن الشرح التلقائي للصور للحد من تكلفة الشرح والفجوة بين دلالية المفاهيم والميزات الرقمية ذات المستوى المنخفض. مساهمتنا في هذا المجال تتجلى في توفير نظام أكثر كفاءة لشرح الصور تلقائيا.

لتحقيق هذا الهدف، قمنا، في البداية، بفحص و استعراض الدراسات المنجزة حول الأساليب المختلفة للشرح التلقائي للصور. وقد سمحت لنا ومكنتنا هذه الدراسة من تصميم بنية نظام الشرح التلقائي الذي تم إنشاؤه. ثم لتنفيذ هذا النظام، قمنا ببرمجة العديد من الخوارزميات لتقسيم الصورة واستخراج واصفاتها و تصنيفها. وقد تم برمجة هذه الخوارزميات على شكل وحدات تسمح لتقنيات مختلفة لتحل محل بعضها البعض من أجل اختيار أنسب مجموعة. لتحسين تشغيل ذالك النظام، تم إجراء دراسات تجريبية من الشرح التلقائي للصور على ثلاث قواعد لبيانات الصور. ولقد سمح لنا تحليل نتائج هذه الدراسات بتحسين البنية الوظيفية المعتبرة لنظام شرح الصور. وتم تعريف البنية الأكثر كفاءة من خلال:

- الجمع بين مقاربتين مختلفتين ومتكاملتين للتصنيف وهما المقاربة التمييزية والمقاربة التوليدية. وقد أدى إجراء الدراسات التجريبية إلى الجمع بين المصنف التمييزي (الشبكة العصبية) مع المصنف التوليدي (شبكة بايز أو شبكة النظرية الافتراضية).

- الجمع بين واصفات الصورة: لقد أظهرت الاختبارات التي أجريناها أن الجمع بين واصفات الصورة أكثر فعالية من دمجها، ويجب أن تستخدم واصفات شكل ورونق ولون الأشياء المزمع شرحها. لذلك قمنا بالجمع بين واصف الشكل، عزم ليجيندر، ورونق الصورة، مصفوفة التشارك، واللون، المدرج الاحصائي للون الأحمر والأصفر والأزرق.

- التجميع بين المناطق المجاورة، الناتجة عن التجزئة التلقائية للصور، للحصول على أشياء مدمجة دلاليا.

هذه البنية الفريدة من نوعها تسمح و تمكن من:

- إمكانية إدماج واصفات أخرى لزيادة و تحسين أداء نظام الشرح الآلي للصور.

- إمكانية إدماج وحدات أخرى لوصف الشيء بواسطة لونه، ورونقه أو حجمه، ... الخ. في هذا العمل، أضفنا لنظام الشرح التلقائي، وحدة لتحديد اللون المهيمن للشيء.

- أخذ القرار بتصويت جميع المصنفات. تعتبر الكلمة ذات أغلبية الأصوات الكلمة الأكثر احتمالا لشرح الشيء.

وتم استخدام هذا النظام التلقائي لشرح صور أخذت من ثلاث قواعد بيانات الصور وهي ETH-80، COIL-100، وقاعدة بيانات شيدت أثناء هذا العمل سميت NATURE. كانت النتائج التي تم الحصول عليها مرضية للغاية.

للذهاب إلى نطاق أوسع، أي إلى قواعد بيانات أكبر تحتوي على صور معقدة، يجب علينا تحسين تقسيم الصور ودمج واصفات أخرى وتغيير لغة البرمجة.

يجب الإشارة أيضا إلى أننا طبقنا الخوارزميات التي تم تنفيذها في هذا العمل لتقسيم وتصنيف الصور واستخراج واصفاتها من أجل الاعتراف التلقائي بأحرف تيفيناغ.

العنوان: الشرح التلقائي للصور.

كلمات البحث: صورة، تجزئة، واصفات، التعلم، الاعتراف، التصنيف، الشبكات العصبية، شبكات النظرية الافتراضية، الشرح، الفجوة الدلالية.

LISTE DES PUBLICATIONS ET COMMUNICATIONS

Publications :

1) *Mustapha OUJAOURA, Brahim MINAOUI and Mohammed FAKIR, Color, texture and shape descriptor fusion with bayesian network classifier for automatic image annotation, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 4, No. 12, pp. 22–29, 2013. Published by The Science and Information Organization, New York, USA.*

2) *Mustapha OUJAOURA, Brahim MINAOUI and Mohammed FAKIR, Multilayer Neural Networks and Nearest Neighbor Classifier Performances for Image Annotation, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 3, No. 11, pp. 165–171, 2012. Published by The Science and Information Organization, New York, USA.*

3) *Mustapha OUJAOURA, Brahim MINAOUI and Mohammed FAKIR. Article: Image Annotation using Moments and Multilayer Neural Networks. (IJCA) International Journal of Computer Applications, Special Issue on Software Engineering, Databases and Expert Systems SEDEX(1): pp. 46-55, September 2012. Published by Foundation of Computer Science, New York, USA.*

4) *M. OUJAOURA, B. MINAOUI, M. FAKIR, B. BOUIKHALENE, R. EL AYACHI and O. BENCHAREF, Invariant Descriptors and Classifiers Combination for Recognition of Isolated Printed Tifinagh Characters, (IJACSA) International Journal of Advanced Computer Science and Applications, special issue on selected papers from third international symposium on automatic amazigh processing SITACAM13, Vol. 3, No. 2, pp. 22–28, 2013. Published by The Science and Information Organization, New York, USA.*

5) *M. OUJAOURA, R. EL AYACHI, M. FAKIR, B. BOUIKHALENE, and B. MINAOUI, Zernike Moments and Neural Networks for Recognition of Isolated Arabic Characters, International Journal of Computer Engineering Science (IJCES), Volume 2 Issue 3, pp.17–25, March 2012.*

6) *Mustapha OUJAOURA, Brahim MINAOUI and Mohammed FAKIR. Article: Walsh, Texture and GIST Descriptors with Bayesian Networks for Recognition of Tifinagh characters. (IJCA) International Journal of Computer Applications, Volume 81– No. 12, pp. 39-46, November 2013. Published by Foundation of Computer Science, New York, USA.*

7) *M. OUJAOURA, B. MINAOUI, M. FAKIR, R. EL AYACHI and O. BENCHAREF, Recognition of Isolated Printed Tifinagh Characters, (IJCA) International Journal of Computer Applications, Volume 85 – No. 1 , pp.1 – 13 , January 2014. Published by Foundation of Computer Science, New York, USA.*

Communications:

1) *Oujaoura, M.; Minaoui, B.; Fakir, M., "A semantic approach for automatic image annotation," Intelligent Systems: Theories and Applications (SITA), 2013 8th International Conference on , vol., no., pp.1–8, 8-9 May 2013, doi: 10.1109/SITA.2013.6560800, ©2013 IEEE.*

2) *Mustapha OUJAOURA, Brahim MINAOUI and Mohammed FAKIR. Image Annotation using Moments and Multilayer Neural Networks. Software Engineering, Databases and Expert Systems (SEDEXS'12), Settat, Morocco, 14-15 June 2012.*

3) *Bencharef, O.; Fakir, M.; Minaoui, B.; Hajraoui, A.; Oujaoura, M., "Color objects recognition system based on artificial neural network with Zernike, Hu & Geodesic descriptors," Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2012 6th International Conference on , vol., no., pp.338–343, 21-24 March 2012, doi: 10.1109/SETIT.2012.6481938, ©2012 IEEE.*

4) *Mustapha OUJAOURA, B. MINAOUI, M. FAKIR, B. BOUIKHALENE, R. EL AYACHI and O. BENCHAREF, Invariant Descriptors and Classifiers Combination for Recognition of Isolated Printed Tifinagh Characters, third international symposium on automatic amazigh processing SITACAM13, Beni Mellal, Morocco, 2-4 May 2013.*

5) *Mustapha OUJAOURA, R. EL AYACHI, B. MINAOUI, M. FAKIR and B. BOUIKHALENE, Zernike Moments and Neural Networks for Recognition of Isolated Arabic Characters, 4th International Conference on Arabic Language Processing, Rabat, Morocco, 2–3 May 2012.*

6) *Mustapha OUJAOURA, Brahim MINAOUI and Mohammed FAKIR. Combined descriptors and classifiers for automatic image annotation, The 4th International Conference on Multimedia Computing and Systems (ICMCS'14), April 14-16 2014, Marrakesh, Morocco.*

LISTE DES FIGURES

LISTE DES TABLEAUX

LISTE DES ALGORITHMES

LISTE DES ABREVIATIONS

ACP	: Analyse en Composantes Principales;
BHDT	: Binary Hierarchical Decision Trees;
BNPC	: Bayes Net Power Constructor;
CBIR	: Content Based Image Retrieval (Recherche d'images par le contenu);
CIE	: Commission Internationale de l'Eclairage;
DAG	: Directed Acyclic Graph;
DDAG	: Decision Directed Acyclic Graphs;
EAP	: Espérance À Posteriori;
EM	: Expectation Maximisation ou Espérance Maximisation;
FST	: Faculté des Sciences et Techniques;
GIF	: Graphics Interchange Format;
GS	: Greedy Search;
GUI	: Graphical User Interface;
HSV	: Hue-Saturation-Value;
IC	: Inductive Causation;
JPEG	: Joint Photographic Experts Group;
K-NN	: K-Nearest Neighbour;
LDA	: Latent Dirichlet Allocation;
LSA	: Latent Semantic Analysis;
MAP	: Maximum À Posteriori;
MWST	: Maximal Weight Spanning Tree.
NTSC	: National Television System Committee;
PAL	: Phase Alternating Line;
PC	: Peter & Clark;
PDF	: Portable Document Format;
PLSA	: Probabilistic Latent Semantic Analysis;
PNG	: Portable Network Graphics;
PSD	: PhotoShop Document;
RBF	: Radial Basis Function;
RGB	: Red-Green-Blue;
RNA	: Réseaux de Neurones Artificiels;
RVB	: Rouge-Vert-Bleu;
SECAM	: SÉquentiel Couleur À Mémoire;
SSL	: Semi Supervised Learning;
SVM	: Support Vector Machines;
TANB	: Tree Augmented Naive Bayes;
TIAD	: Traitement de l'Information et Aide à la Décision ;
TIT	: Traitement de l'Information et Télécommunications;

TABLE DES MATIERES

Chapitre I INTRODUCTION GENERALE

"I have not failed 700 times. I have not failed once. I have succeeded in proving that those 700 ways will not work. When I have eliminated the ways that will not work, I will find the way that will work".

[Thomas Edison, about creating the light bulb]

Contenu du Chapitre

I.1 Contexte et motivations

A la différence des autres créatures et étant doté d'une très complexe machine biologique à penser et raisonner, l'être humain ne cesse d'exploiter son environnement afin de découvrir de nouvelles méthodes et techniques qui lui facilitent la vie ou tout simplement pour combler la soif qu'il approuve vis-à-vis de la connaissance. Après l'automatisation de beaucoup de processus dans la plupart des domaines, l'être humain a pu commencer l'automatisation du traitement de l'information en développant le premier calculateur vers les années 50. Cette soif et cette volonté d'avoir et d'acquérir de plus en plus de connaissance pour une raison ou une autre (guerre mondiale, guerre froide, vie confortable, conquête spatiale, crise économique, anti-terrorisme, etc.…) a permis aux êtres humains de continuer à inventer et innover. Des avancées spectaculaire ont été connues et vécues dernièrement dans tous les domaines, plus particulièrement, dans le domaine de traitement automatique des informations.

Actuellement, les réseaux sociaux comme Facebook, Google+, Twitter, Linkedin comptent des millions d'utilisateurs de divers nationalités qui interagissent et échangent plusieurs formes d'informations. Le 21ème siècle est alors l'ère de l'information par

excellence, au cours duquel l'organisation et l'indexation des informations de manière plus efficace est d'une importance essentielle. En effet, avec le développement rapide de l'imagerie numérique, la façon de rechercher des images plus efficacement par leur contenu est devenue l'un des plus grands défis. La Recherche d'images par le contenu (Content Based Image Retrieval : CBIR) a été la technique traditionnelle et dominante pour la recherche d'images depuis des décennies [***1***] [***2***]. Cependant, ce n'est que récemment que les chercheurs ont commencé à prendre conscience des problèmes vitaux qui existent dans les systèmes de recherche d'images par le contenu (CBIR). L'un des plus importants problèmes est connu sous le nom de fossé sémantique (Semantic Gap en anglais), qui fait référence à l'écart entre l'information qui peut être extraite à partir d'images et l'interprétation humaine des images. Pour tenter de combler ce fossé sémantique, l'annotation d'images a gagné de plus en plus d'attention au cours des dernières années en parallèle avec la recherche d'images. Compte tenu de la croissance rapide du nombre d'images numériques, l'annotation manuelle prend beaucoup de temps et dépend de l'intervenant qui fait l'annotation même s'il est expert dans ce domaine. L'automatisation de cette lourde tâche peut résoudre au moins partiellement ce problème.

L'annotation automatique d'images est le calcul de métadonnées textuelles qui représentent et décrivent le contenu des images (légende et mots-clés). Il s'agit de la découverte de correspondances entre les caractéristiques visuelles des images et des mots textuelles, le plus souvent grâce à des techniques de modélisation probabiliste, d'apprentissage et de classification automatiques. Etant donné que l'annotation d'images constitue l'un des principaux moyens d'indexation d'images, l'annotation automatique d'images ainsi que la recherche d'images basées sur le contenu (CBIR) ont été étudiées toutes les deux depuis plus d'une décennie. Mais en raison du grand écart sémantique persistant, les progrès dans ce domaine sont encore limités. Dans un large volume de données contenant des centaines ou quelques milliers de classes, les défis de l'annotation d'images se trouvent principalement dans la construction de bons modèles qui intègrent les domaines de connaissances et se généralisent ainsi sur de nouvelles données. Dans un large domaine d'informations et de connaissances tel que l'Internet, où les propriétés d'images et les sujets varient considérablement à travers le spectre de représentation d'information, les méthodes d'annotations automatiques d'images peuvent servir pour contribuer à l'efficacité des moteurs de recherches multimédia et faciliter ainsi la recherche dans l'espace des caractéristiques visuelles à grande échelle.

L'annotation automatique des images est un sujet d'actualité au sein de la communauté de d'indexation et de recherche d'images. Actuellement, plusieurs images des bases de données comme Flickr sont soit annotées de façon incorrecte, soit annotées partiellement ou ne sont pas encore annotées. En plus du fait que l'annotation manuelle d'un large volume de données est fastidieuse et inefficace, les performances des systèmes actuels, qui exploitent majoritairement les caractéristiques de bas niveau des images (couleur, texture, forme), ne sont pas encore satisfaisants. D'où l'importance d'un processus d'annotation automatisé afin de réduire l'intervention humaine et les défauts qui lui sont liés. Il est nécessaire alors de proposer des améliorations pour l'annotation automatique d'images. Le principe envisagé pour ceci est d'utiliser conjointement plusieurs descripteurs et classificateurs permettant une analyse du contenu d'images afin d'améliorer l'efficacité des systèmes d'annotations automatiques d'images. D'où le grand intérêt de la recherche présentée dans cette thèse qui a pour sujet **l'annotation automatique d'images**. L'amélioration, d'une manière ou d'une autre, des différentes phases d'annotation automatique d'images permettra sans aucun doute l'annotation performante du contenu visuel d'images couleurs.

I.2 Objectifs et contributions

Cette thèse vise à explorer un certain nombre de différentes approches d'annotation automatique d'images et certaines questions connexes qui constituent le fondement de la recherche dans ce domaine. L'objectif est de fournir les différentes techniques d'annotation sémantique du contenu visuel d'images afin d'améliorer le taux d'annotation d'images et réduire ainsi le fossé sémantique d'images couleur qui est le principal obstacle de la recherche d'images par contenu visuel (CBIR). Ce travail de recherche a pour but de développer un système pour extraire une signification sémantique des images couleurs. Étant donné une image d'entrée, notre système va générer automatiquement des mots-clés qui décrivent son contenu visuel. Le système est mis au point sur un ensemble de bases de données d'images.

Pour améliorer le taux d'annotation, nous avons adopté une approche basée sur la segmentation d'une image en plusieurs objets afin de les classifier et par la suite leur attribuer des mots clés convenables. Nous avons alors basé notre contribution sur la segmentation, la description et la classification du contenu d'images qui sont des tâches importantes en traitement d'images en essayant de les améliorer et de les adapter au problème d'annotation.

Du fait que nous avons remarqué que certaines méthodes de classification et caractérisation du contenu d'images sont performantes pour un certain type de classe d'images et d'autres méthodes pour d'autres types de classes d'images, nous avons jugé qu'il est très utile de proposer une approche d'annotations combinant plusieurs techniques de description et de classification d'images. Du fait aussi que la plupart des méthodes de segmentations d'images utilisent des prédicats de bas niveau pour contrôler l'homogénéité des régions résultantes, ces régions ne sont pas tout à fait compactes sémantiquement et ne représentent pas réellement les objets contenu dans une image. Afin de remédier à ce problème, nous avons aussi proposé une méthode basée sur le regroupement des régions adjacentes pour augmenter leur degré de compacité sémantique et améliorer ainsi le taux du système d'annotation automatique du contenu visuel d'images couleurs.

I.3 Structure et articulation de la thèse

Le présent mémoire se compose de six chapitres et deux annexes dont quatre chapitres sont regroupés en deux parties essentielles : la première partie comporte un état de l'art concernant le sujet de la thèse tandis que la deuxième partie comporte la conception et la réalisation ainsi que les résultats du système d'annotation automatique d'images que nous avons réalisé. Elle est débutée par une introduction générale dont nous avons présenté le contexte et les motivations de recherche du sujet d'annotation automatique du contenu visuel des images couleurs ainsi que les objectifs et les contributions de cette thèse. Le mémoire se termine par une conclusion générale et une perspective des travaux de recherches réalisés.

Le chapitre II est consacré à la définition des différents types d'annotation d'images. Il traite et présente également la problématique et l'état de l'art concernant les systèmes d'annotation d'images.

Dans le chapitre III, une brève présentation de la structure globale du système d'annotation automatique d'images est donnée également ainsi qu'une brève description des différentes parties dont il se compose. Enfin, les différents critères d'évaluation des performances d'un tel système sont présentés.

Le chapitre IV est réservé à la conception du système d'annotation automatique du contenu visuel d'images. Il expose une étude détaillée des différentes parties du système d'annotation automatique proposé, et ceci en présentant le formalisme mathématique et

algorithmique nécessaires pour une telle étude. Nous avons commencé par une présentation de la structure générale du système d'annotation automatique d'images qui se compose de trois phases essentielles, à savoir : la segmentation d'images, l'extraction d'attributs d'images et l'annotation par classification d'images. Ensuite nous avons étudié et présenté deux méthodes différentes de segmentation d'une image couleur en objets sous forme de régions. Pour la deuxième phase du système d'annotation, plusieurs techniques d'extraction de caractéristiques d'objets contenus dans une image sont présentées en détail. Enfin pour la dernière phase du système d'annotation, certains classificateurs ont été proposés afin de classifier les objets pour les annoter par des mots clés convenables.

Dans le chapitre V, la réalisation et la mise en œuvre du système d'annotation automatique d'image conçu est présentée. Les différents résultats sont présentés et discutés pour différentes bases de données d'images afin de faire une évaluation objective des approches proposées.

Enfin, dans le chapitre VI, Nous concluons ce mémoire en soulignant les résultats importants obtenus, en rappelant nos contributions, et en dégageant les perspectives principales de ce travail.

Partie 1 : ETAT DE L'ART & STRUCTURE GENERALE DES SYSTEMES D'ANNOTATION D'IMAGES

Chapitre II ETAT DE L'ART

« Une image vaut mille mots »

[Le philosophe chinois Confucius]

Contenu du Chapitre

II.1 Introduction

Le but de ce chapitre est de présenter un aperçu de la recherche qui est liée à cette thèse, en mettant l'accent sur les techniques d'annotation automatique d'images les plus récentes. Dans ce chapitre, nous allons d'abord commencer par définir et spécifier les différents types d'annotations d'images et nous allons présenter, par la suite, la problématique et un état de l'art sur l'annotation automatique du contenu visuel des images couleurs que ce soit l'annotation manuelle, l'annotation automatique ou l'annotation semi-automatique.

II.2 Définition et types d'annotations d'images

L'annotation d'images est un commentaire, une note, une explication ou toute autre remarque sous forme textuelle qui peut être attachée à une image ou à une partie de celle-ci. D'un point de vue technique, une annotation peut être vue comme une métadonnée puisqu'elle fournit une information supplémentaire sur une donnée existante. L'annotation d'images peut être alors définie comme étant le calcul de métadonnées textuelles pour les

images (légende et mots-clés). Il s'agit de la découverte des correspondances entre les caractéristiques visuelles des images et des mots textuelles.

Dans la plupart des cas, du point de vue moyens utilisés, on distingue trois types d'annotations qui sont l'annotation manuelle, l'annotation automatique et l'annotation semi-automatique. Du point de vue niveaux d'intervention, on distingue deux types d'annotations [***1***]: le premier type qui décrit ce que l'on voit dans l'image et le second type qui exprime la signification et le contexte de l'image. On peut trouver aussi les types d'annotations suivantes :

- **Globales** : elles décrivent le type d'images (photo, graphique, croquis, image de synthèse, …) ou elles caractérisent la scène dans son ensemble (intérieur, extérieur, jour, nuit, paysage, ville, portrait, horizontal, vertical, …).
- **Locales** : elles permettent de décrire plus précisément le contenu de l'image et indiquent la présence d'objets ou de personnes.
- **Contextuelles** : elles servent à situer l'image. Ce qu'elles décrivent n'apparaît pas directement sur l'image mais permet d'indiquer le lieu, la date ou l'auteur et de décrire l'événement qui est pris en photo.
- **Subjectives** : elles évoquent par exemple des émotions que l'image est supposée provoquer (douceur, tristesse, colère, …).
- **Techniques** : telles que les réglages d'un appareil photo ou les caractéristiques des différents éléments d'une chaîne de numérisation.

Actuellement, les appareils photos numériques sont capables d'ajouter automatiquement un certains nombre d'annotations tels que les réglages techniques de l'appareil, la date de la prise de vue, le nom du photographe, le sens de prise de vue (horizontal ou vertical), une localisation géographique avec un GPS et même l'indication de présence de visages sur la photo. En dehors de ces annotations techniques, des mots-clés ou bien des phrases complètes sont généralement ajoutées manuellement. Pour le cas particulier des images sur Internet, le texte entourant une image sur une page web est souvent utilisé pour la décrire. Néanmoins, la plupart des images ne sont pas alors complètement annotées. Dans cette thèse, nous nous intéresserons seulement à l'annotation automatique locale d'images du fait qu'elle a comme objectif la description précise du contenu visuel d'une image.

II.3 Intérêt et problématique de l'annotation d'images

Avant d'entamer toute étude sur le système d'annotation automatique d'images couleurs, il serait bien utile de mettre en évidence son intérêt et son utilité pour le traitement d'information en général et en particulier pour le traitement d'images, surtout pour l'indexation et la recherche d'images. Ceci permettra de découvrir et de dégager grosso modo la problématique de l'annotation automatique du contenu visuel d'images en couleurs.

II.3.1 Intérêt et objectif de l'annotation d'images

Contrairement aux textes qui nécessitent du temps pour être lus et déchiffrés, les images sont impressionnantes du fait que le contenu d'une photo peut être saisi et décrypté rapidement. Cette instantanéité de perception et la confiance portée généralement au contenu visuel incitent peu à l'analyse et à la prise de recul. Les photos ont ainsi la capacité de faire surgir immédiatement plusieurs sentiments chez ceux qui les regardent : attrait, indifférence, désir, nostalgie, etc.… Pourtant, une photo ne décrit pas une unique réalité, mais une réalité telle qu'elle est perçue par un être humain à partir d'un contexte historique et social. De nos jours, les images sont présentes partout. Leur influence dans nos sociétés est souvent primordiale. Elles sont utilisées pour témoigner de l'actualité, pour illustrer les articles de journaux. Dans tous les cas, il est important de savoir lire une image, de décoder et déchiffrer ses différents mécanismes. C'est un long apprentissage, mais il est important.

Les progrès technologiques récents en matière d'acquisition et de diffusion de données multimédia ont conduit à une croissance exponentielle du nombre d'images disponibles. On retrouve maintenant des bases d'images dans tous les domaines de la société. Le terme image regroupe tout contenu visuel statique. Outre les photographies classiques, il peut s'agir de dessins, tableaux, schémas ou encore d'images scientifiques. Leur seul point commun est d'être sous format numérique. On distingue deux types de bases d'images [***2***]. Les bases généralistes ont une grande variabilité du contenu qui peut atteindre plusieurs dizaines de milliards d'images. En plus de l'acquisition et la diffusion toujours plus rapide de nouveaux contenus, d'autres fonds d'archives sont en cours de numérisation. A côté de ces bases généralistes, on trouve de très nombreuses bases spécifiques dans des domaines d'applications précis. On peut citer les bases scientifiques (biologie, médecine, astronomie, …), satellitaires (cartographie, météo, agriculture, militaire, …), biométriques (visages, empreintes digitales, iris, …) ou encore les archives culturelles (œuvres d'art, numérisation des livres, …).

La navigation et la recherche d'informations dans ces bases est une activité cruciale et primordiale pour leurs utilisateurs. Traditionnellement, des moteurs de recherche de textes classiques sont utilisés pour la recherche d'images annotées manuellement. Les approches et les algorithmes utilisés dans ce cas sont directement issus du monde du traitement du langage naturel. En plus des problèmes liés d'une part à tous les inconvénients liés aux moteurs de recherche texte et inhérents à la langue (polysémie, multilinguisme, synonymies, …) et à la formulation des requêtes, la recherche d'images fait face, d'autre part, au second type de problèmes qui sont directement liés à la nature des images et leurs annotations. En effet, la subjectivité de l'opérateur humain lors de l'annotation entre en compte. Deux personnes n'attribueront probablement pas les mêmes mots-clés ou les mêmes descriptions pour une image donnée [***3***], [***4***], [***5***]. Au delà de la subjectivité des personnes qui annotent les images, il faut également tenir compte de la façon dont une même image peut être interprétée dans des contextes ou des cultures différentes. Une autre possibilité pour tenter de résoudre ce problème est de faire en sorte que plusieurs personnes annotent une même image. L'annotation manuelle est alors une opération coûteuse en temps et qui ne garantit pas une satisfaction totale.

Face à ce constat, un nouveau domaine de recherche a fait son apparition : la recherche d'images par le contenu (Content Based Image Retrieval en anglais, CBIR). Le but est de se baser directement sur le contenu visuel des images et sur leur analyse pour naviguer et effectuer des recherches dans les bases de données d'images. La recherche par le contenu visuel permet de compenser certains défauts des descriptions textuelles. Elle s'est révélée efficace et très utile dans de nombreux domaines d'application. Toutefois, cette approche, qui est basée sur la correspondance entre les caractéristiques de bas niveau des images telles que la couleur, la texture la forme d'une part et les requêtes d'autre part, possède également ses propres limitations. En effet, la machine qui peut être un ordinateur stocke les images sous une forme à base de pixels et les humains utilisent des descriptions sémantiques pour les décrire. La Figure II-1 montre la pyramide des différents niveaux de représentation de l'image.

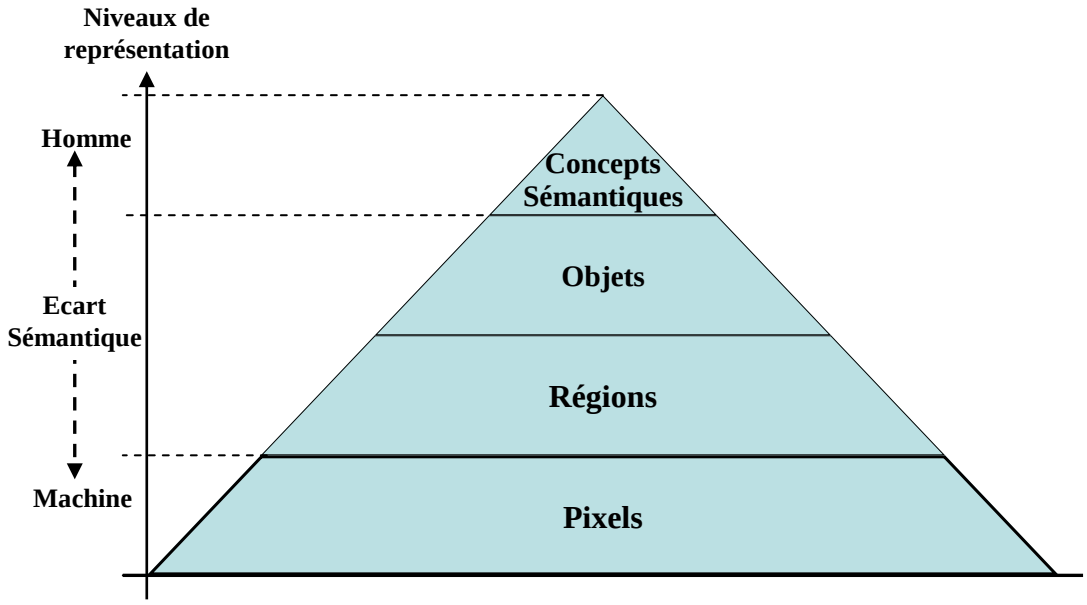


Figure II-1 : Pyramide des différents niveaux de représentation de l'image.

La représentation de l'image en fonction des caractéristiques de bas niveau est difficile à comprendre pour les gens [***6***]. Le principal défi de la recherche d'images par le contenu est de combler le fossé sémantique existant entre les caractéristiques de bas niveau et le contenu conceptuel des images [***2***], [***7***], [***8***]. Pour la recherche de l'image, l'expression de la requête d'images par mots-clés est plus adaptée que par des caractéristiques de bas niveau, parce que les gens décrivent souvent une image avec des mots, pas avec des fonctionnalités de bas niveau [***9***].

Il apparaît alors que la combinaison des deux sources d'informations, textuelle et visuelle, est primordiale pour augmenter l'efficacité de l'interrogation des bases d'images [***10***], [***11***]. D'autre part, l'annotation automatique propose d'apprendre des modèles pour un certain nombre de concepts visuels. Ces modèles sont ensuite utilisés pour prédire la présence des concepts sur les images et générer ainsi de nouvelles annotations. Elle permet de soulager l'utilisateur d'une partie du travail fastidieux. Il faut toutefois bien garder à l'esprit que le but principal est de permettre une recherche efficace dans les bases d'images. L'exactitude des annotations générées, leur cohérence et leur utilité sont donc des critères importants dont il faut tenir compte.

Les détenteurs professionnels de contenu visuel doivent adapter leur processus de travail pour des utilisations sur Internet qui nécessitent une forte réactivité dans la mise à disposition du contenu. Pour les images d'actualité, plus le contenu est en ligne rapidement,

plus il a de chance d'être visité et vendu. Les systèmes doivent donc intégrer ces contraintes tout en maintenant leurs objectifs de qualité. Le problème est que certaines agences photo ont tendance à sur-annoter accidentellement ou volontairement leurs contenus avec de très nombreux mots-clés redondants, souvent sans réel lien avec l'image, en espérant ainsi que les images soient retournées comme résultats pour différentes requêtes. Le contenu est souvent diffusé en n'étant que très peu annoté dans un premier temps ou même sur-annoté pour des fins commerciales. Ceci constitue une mauvaise pratique de l'annotation par erreur ou pour des fins commerciales. Pour résoudre ce problème, les moteurs de recherches doivent avoir leurs propres systèmes d'indexation et d'annotation d'images.

Quand aux photos prises par les particuliers, de façon classique, les photos numériques sont stockées et archivées sur des supports de stockage numériques (ordinateur, cartes mémoires ou cd-rom) ou éventuellement imprimées pour être conservées dans un album traditionnel. La consultation des archives subsiste dans le cadre familial. Une étude sur la façon d'organiser des photos personnelles précise que les utilisateurs sont relativement peu intéressés par des fonctionnalités d'annotation de leurs images puisqu'ils connaissent leurs photos et que l'effort nécessaire est trop grand et demande beaucoup de temps [***12***]. De plus, l'apparition et le succès phénoménal des réseaux sociaux (Facebook, Google+, Twitter, Linkedin) et des sites de partages de photos (Flickr) fait émerger de nouveaux besoins. Des milliards d'images sont disponibles et qui ne cessent de croître chaque jour. Les usages sont sensiblement différents entre Flickr et Facebook. L'idée maîtresse du premier (Flickr) est le partage des photos et de faire en sorte qu'elles soient vues par le plus grand nombre d'internautes. L'annotation et le classement des photos sont favorisés en plus de la qualité esthétique des images qui est souvent mise en avant [***13***], [***14***]. Sur Facebook en revanche, l'hébergement de photos est principalement tourné vers l'identification des personnes présentes sur les clichés. La consultation de ces images est généralement plus restreinte et réservée au cercle des relations proches. Le fait que les collections de photos personnelles sortent du cadre strictement familial pour être accessibles via Internet rend plus que jamais nécessaires les outils pour parcourir et chercher dans ces masses de données. En effet, les collections de photos n'étant plus réservées à leurs seuls auteurs, leur connaissance n'est plus assurée pour les personnes qui les consultent. En revanche, l'effort nécessaire à l'annotation manuelle étant toujours présent, les outils d'annotation automatique sont promis à un bel avenir sur ce type de sites pour peu qu'ils réussissent à répondre à des besoins concrets.

Un moteur de recherche ne peut travailler qu'à partir de l'information dont il a connaissance, c'est- à-dire l'ensemble des métadonnées qui auront été fournies ou extraites des images par l'annotation [***15***]. Il importe donc de mettre l'accent sur l'extraction de ces annotations puisqu'elles sont d'un intérêt capital pour la recherche d'images. L'utilisateur exprime ses requêtes de façon pauvre. Il y a généralement un gap assez large entre l'intention de l'utilisateur et la manière dont il l'exprime, ou encore la manière dont le système le comprend. Réduire ce gap est l'un des objectifs principaux que les moteurs de recherche doivent atteindre. Ce problème est potentiellement plus difficile pour les moteurs multimédia que pour les moteurs textes. La force d'un bon moteur de recherche réside dans le degré d'assistance fournie à l'utilisateur.

Toutefois, il n'est pas préférable et opportun de distinguer les annotations textuelles des métadonnées qui sont liées à l'analyse du contenu visuel. Toutes ces métadonnées sont des moyens d'accéder au contenu en utilisant le moteur de recherche et les paradigmes de requête adéquats. Le système d'annotation automatique doit être capable de générer un score de confiance ou une probabilité concernant un mot-clé ou un concept visuel. Ce score va pouvoir s'intégrer dans le moteur de recherche au même titre que les autres et pourra servir à naviguer dans la base, à la filtrer, à la catégoriser, …. [***16***], [***17***]. Nous concluons alors que l'annotation d'images constitue une étape cruciale pour l'indexation des bases de données d'images afin de faciliter la tâche de recherche et catégorisation des images.

II.3.2 Problématique de l'annotation d'images

Lors de la définition d'un système de vision par ordinateur comme celui de l'être humain, on peut distinguer généralement trois principaux problèmes à résoudre, chacun selon son importance et sa difficulté : le gap sensoriel, le gap numérique et le gap sémantique [***2***], [***18***].

Le gap sensoriel représente la perte ou la déformation d'information liée à la technicité du capteur (scanner, appareil photo numérique, caméra) lors de la phase d'acquisition d'une image. Les mêmes considérations peuvent être faites pour les appareils de mesure scientifiques produisant également des images (domaine médical, satellitaire, ...). Les problèmes classiques regroupés sous l'appellation de gap sensoriel sont typiquement : la perte d'informations liée à la discrétisation et à la capacité du capteur, les déformations optiques, le bruit numérique, les approximations colorimétriques, ...

Le gap numérique représente la sous-capacité (incapacité) d'un descripteur à extraire des signatures visuelles pertinentes pour un système de vision. Ce problème peut être lié au choix même du descripteur utilisé. Par exemple, pour une tâche donnée, la caractéristique pertinente à analyser est la couleur mais on utilise un descripteur de forme, ou bien encore on utilise un descripteur global alors qu'on ne s'intéresse qu'à des petits détails de l'image où un descripteur local aurait été plus performant. Il peut également être résultant des choix des différents paramètres du descripteur (quantification trop faible, mauvais facteur d'échelle, ...). C'est donc l'écart entre l'information qui est présente visuellement dans une image et celle qu'un descripteur est capable d'extraire et de représenter. Il pose le problème de la fidélité de la signature par rapport à l'image.

Enfin, le gap sémantique représente l'écart entre la sémantique qu'un système de vision cognitive est capable d'extraire pour une image et celle qu'un utilisateur aura pour cette même image. Le principal problème réside dans le fait que toute image est contextuelle. L'ensemble des informations situant ce contexte (géographique, temporel, culturel, ...) ne sont pas présentes visuellement dans l'image et font appel à la mémoire de l'utilisateur.

Le but est donc de réduire ces différents gaps. On peut remarquer que le gap sensoriel n'est, théoriquement, pas forcément un problème puisqu'un utilisateur à qui on présente une image numérique est presque toujours capable d'en comprendre le sens, indiquant ainsi que l'information visuelle contenu dans l'image est suffisante. Le gap sémantique est souvent mis en avant comme étant la seule explication à la difficulté de concevoir un système ayant de bonnes performances. Il ne faut toutefois pas négliger les deux premiers gaps. L'adéquation d'éventuels prétraitements et des descripteurs avec la tâche à effectuer est primordiale dans les performances.

Jusqu'à présent, pour un système d'annotation automatique d'images, le problème fondamental de l'écart sémantique ou encore fossé sémantique existe toujours quand un très grand nombre de mots-clés doivent être affectés aux images. Autrement dit, les caractéristiques extraites de bas niveau sont très susceptibles d'être visuellement similaires mais sémantiquement différents ou ils sont visuellement dissemblables mais sémantiquement similaires. Les approches utilisées dans la littérature sont encore limitées et ne sont pas en mesure de résoudre totalement ce problème persistant. Ceci est dû principalement au fait que les algorithmes d'apprentissage actuels ne sont pas encore capables d'apprendre à travers des

représentations d'entités de bas niveau pour reconnaître des mots-clés conceptuelles de haut niveau convenables à l'annotation d'images.

II.4 Etat de l'art de l'annotation d'images

Nous venons de voir précédemment que l'association d'information sémantique à une image est toujours un défi. En effet, l'indexation textuelle d'images se contente d'indexer les images sur la base de mots-clés représentatifs d'images, mais sans utiliser le contenu de l'image (les caractéristiques visuelles). De même, l'indexation d'images basée sur le contenu organise les images sur la base de caractéristiques visuelles de bas niveau (couleur, texture, forme, ...). Mais aucun lien entre l'information sémantique et l'information visuelle n'est fait à ce niveau. On parle de l'écart sémantique ou encore de fossé sémantique qui est déjà défini (plus connu sous le nom de semantic gap, en anglais) [***2***].

L'annotation d'images constitue une manière possible d'associer de la sémantique à une image, et permet ainsi de réduire le fossé en question. En effet, elle consiste à assigner à chaque image, un mot-clé ou un ensemble de mots-clés, destinés à décrire le contenu sémantique de l'image. Cette opération peut être vue comme une fonction permettant d'associer de l'information visuelle, représentée par les caractéristiques de bas niveau de l'image (forme, couleur, texture, ...), à de l'information sémantique, représentée par des mots-clés.

L'annotation d'images va alors pouvoir être utilisée en amont de la recherche et de la classification d'images. En effet, les annotations pourront être utilisées pour indexer textuellement les images. De ce fait, les annotations serviront à organiser et localiser les images pour améliorer la classification et la recherche visuelles et textuelles d'images.

Les techniques d'annotation d'images peuvent être réparties en 3 catégories [***19***]. On distingue l'annotation automatique d'images, de l'annotation semi-automatique, qui nécessite l'intervention de l'utilisateur pour valider des annotations automatiques, dans un système de retour de pertinence, par exemple. Enfin on compte aussi l'annotation manuelle d'images, qui consiste à faire annoter des bases d'images par un ensemble d'utilisateurs, avec des mots-clés souvent issus d'un ensemble de mots-clés prédéfini. L'annotation manuelle correspond donc à la première phase d'indexation textuelle des images, à savoir l'extraction de mots-clés. L'annotation manuelle, bien que coûteuse, est souvent nécessaire pour créer des vérités-

terrains, dans le cadre de la validation des approches automatiques. Une brève présentation ainsi qu'un état de l'art des trois types d'annotation d'images sont présentés dans ce qui suit.

II.4.1 Annotation manuelle

L'indexation textuelle manuelle par l'extraction de mots-clés est la plus efficace étape d'annotation manuelle. Par contre, cette technique est coûteuse pour l'utilisateur et elle devient très difficile à appliquer sur de grandes bases d'images. De plus, une même image peut être indexée différemment par différents intervenants. Ce phénomène s'avère d'autant plus courant dans le cas de grandes bases d'images généralistes, où les indexeurs, ne sont pas experts dans tous les domaines qui peuvent être représentés dans de grandes bases d'images généralistes. Afin de pallier ces problèmes et de pouvoir annoter et indexer manuellement de grandes bases d'images généralistes de façon correcte, des sites Web d'annotation d'images ont vu le jour.

Par exemple, l'outil d'annotation manuelle en ligne LabelMe [***20***], permet aux utilisateurs de segmenter manuellement les images en régions, puis d'annoter ces régions en choisissant des mots-clés dans une liste. La liste de mots-clés proposée est différente pour chaque image. Elle est établie en fonction des mots-clés déjà choisis par d'autres utilisateurs pour cette image. De cette façon, on limite les erreurs dues à l'ambiguïté des termes. De même, sur certaines images, on peut observer que des régions sont déjà délimitées. Cela veut dire que cette image a déjà été segmentée par un autre utilisateur, et que l'on peut observer les régions qu'il a délimitées. De plus, si une région a déjà été annotée, il suffit de passer la souris dessus pour afficher les mots-clés qui l'annotent. Par exemple, la Figure II-2 montre une image en cours d'annotation avec l'outil LabelMe[1]. Détourées ou entourées en différentes couleurs, on peut observer les régions déjà reconnues par d'autres utilisateurs. Les mots-clés qui ont déjà servi à annoter cette image figurent à droite de l'image. L'utilisateur peut alors segmenter l'image en entourant les régions ou les objets de son choix et les annoter à l'aide des mots-clés fournis ou de nouveaux mots-clés.

[1]Outil LabelMe : http://new-labelme.csail.mit.edu/Release3.0/#content-inner-1

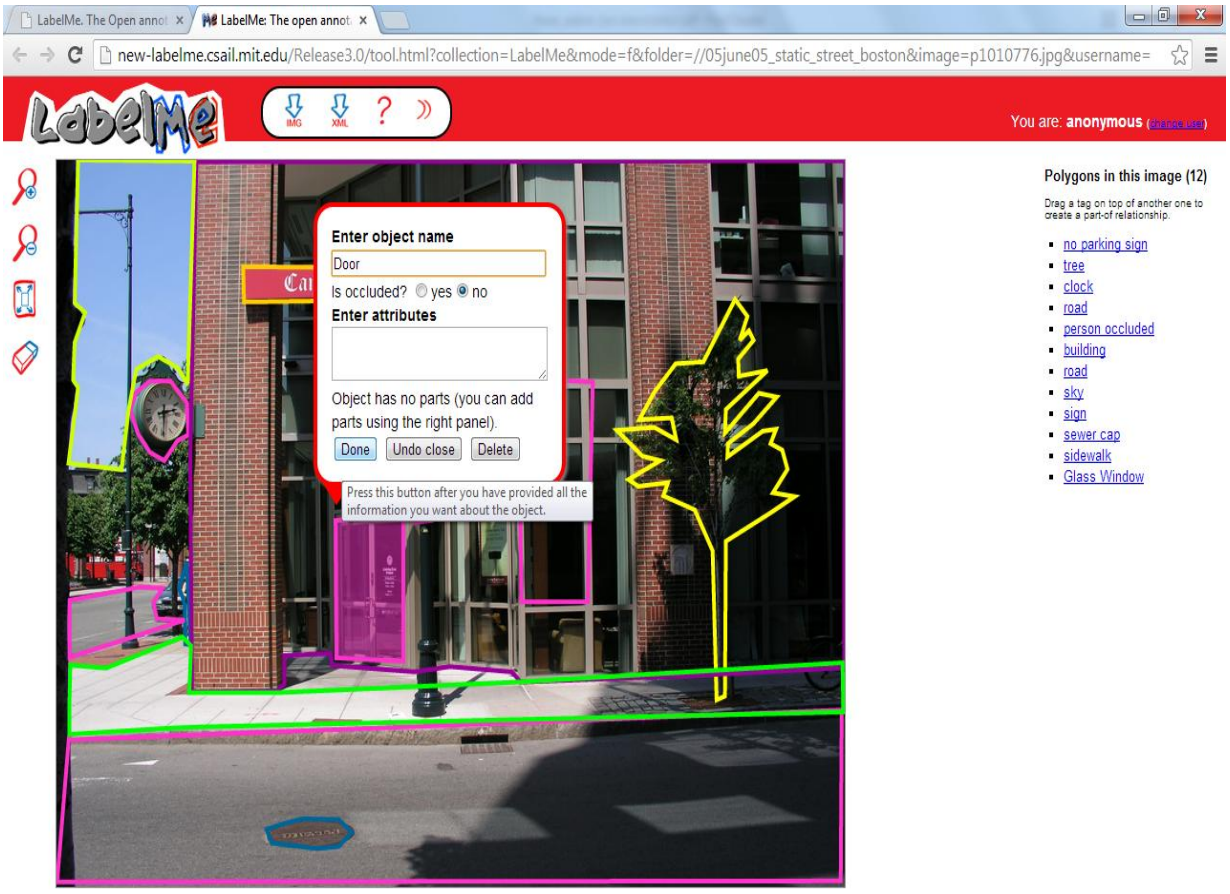


Figure II-2 : Exemple d'annotation d'une image avec l'outil d'annotation d'images LabelMe.

Il existe d'autres outils collaboratifs en ligne comme l'outil Marqueed[2] (voir la Figure II-3) permettant de travailler à plusieurs sur des images qui permettent à différents participants de commenter et d'annoter une même image en sélectionnant les zones d'insertion. Pour cela il suffit d'ouvrir un compte et de glisser-déposer sur la plateforme une ou plusieurs images qui pourront être ensuite partagées avec plusieurs personnes après invitation. Les formats supportés sont les suivants : JPEG, PNG, GIF, PSD, PDF. On peut également capturer une page web entière en soumettant son URL.

[2]Outil Marqueed : https://www.marqueed.com/

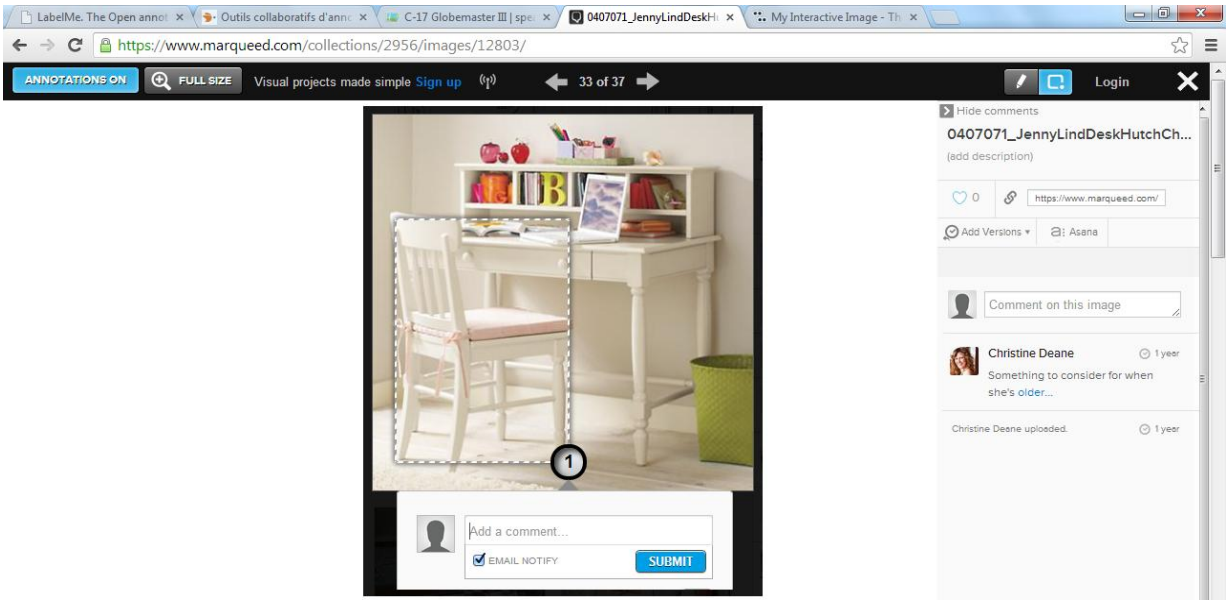


Figure II-3 : Exemple d'annotation d'une image avec le service Web d'annotation d'images Marqueed.

Cet outil (Marqueed) s'inscrit dans la même logique que l'outil SpeakingImage[3], autre application web permettant de créer des images interactives et de les partager avec les autres (voir la Figure II-4). Avec ce service, on a également la possibilité de créer des groupes, d'ajouter des wikis et de définir des autorisations différentes pour gérer le travail collaboratif.

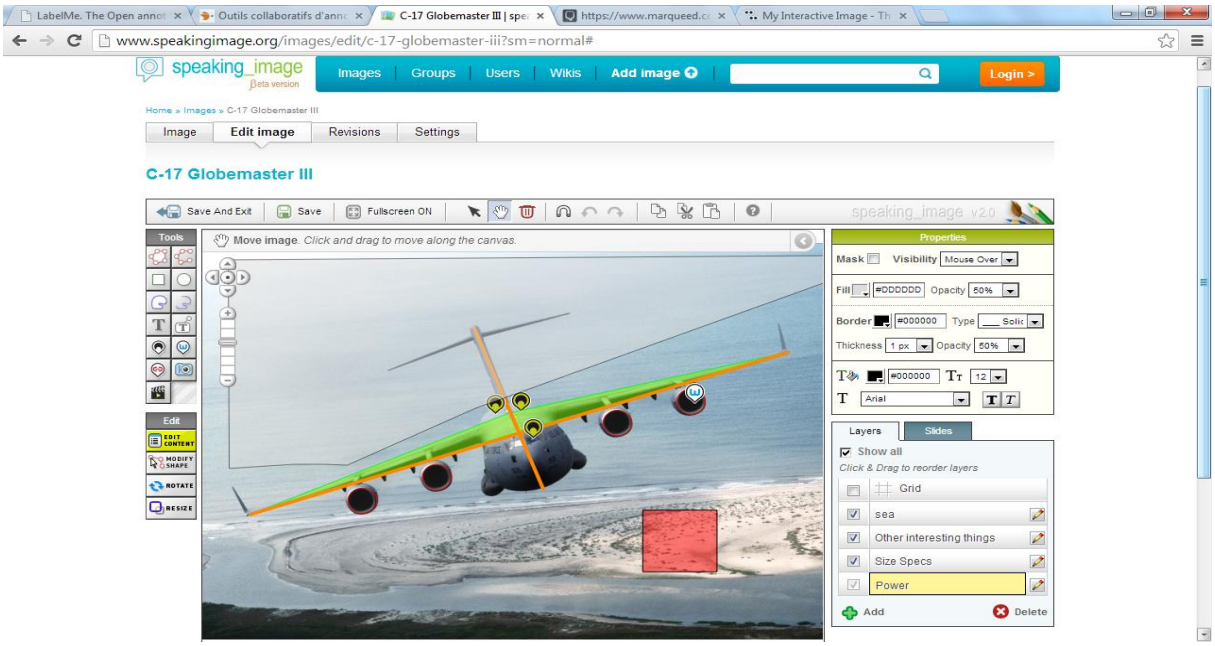


Figure II-4 : Exemple d'annotation d'une image avec le service Web d'annotation d'images SpeakingImage.

[3] Outil SpeakingImage : http://www.speakingimage.org/

La dimension interactive évoquée ci-dessus se retrouve également chez l'outil web Thinglink[4], service en ligne offrant la possibilité d'ajouter des petites infos bulle cliquables à l'image de son choix (voir la Figure II-5).

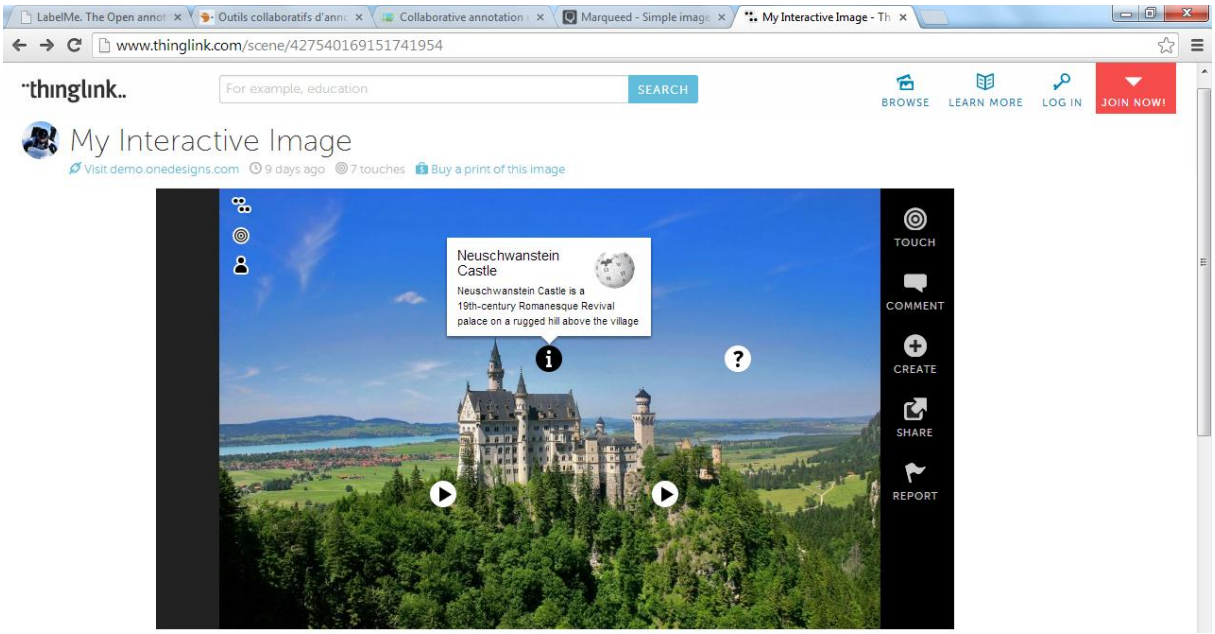


Figure II-5 : Exemple d'annotation d'une image avec le service Web d'annotation d'images Thinglink.

Dans le but de faire oublier à l'utilisateur le côté coûteux de l'annotation, d'autres outils d'annotation en ligne sont présentés sous forme de jeu tel que l'ESP game qui n'est plus disponible et opérationnel sur le Web malheureusement [***21***], [***22***]. Le principe de ce jeu était simple : deux personne voient une même image et lui associent des mots-clés; si un même mot est proposé par les deux : il est validé. La base de données d'images annotées résultante de ce jeu reste encore disponible sur le web[5].

Ces tentatives sous forme de sites web d'annotation d'images sont donc une idée astucieuse pour réduire les erreurs d'annotations liées à la polysémie de l'image. De plus, les sites sous forme de jeu, grâce au charme du gain et leur coté ludique et compétitif (chacun cherche à trouver le même mot que son adversaire, le plus rapidement possible), paraissent être la solution idéale pour motiver les utilisateurs à annoter des images, même dans de grandes bases d'images généralistes. Mais l'annotation de grandes bases de données d'images

[4] Outil Thinglink : http://www.thinglink.com/
[5] Small ESP Game Dataset : http://www.cs.cmu.edu/~biglou/resources/

liées à un domaine particulier ne peut pas être facilitée grâce à ces sites internet et reste la tâche, coûteuse, destinée à des indexeurs expert du domaine.

Pour pallier ce problème de coût de l'annotation qui subsiste malgré ces outils d'annotations, les méthodes d'annotation automatique d'images ont fait leur apparition.

II.4.2 Annotation automatique

L'annotation automatique d'images consiste à associer, à chaque image, un groupe de mots qui décrit le contenu visuel de l'image, au moyen d'un système sans aucune intervention humaine. Cette tâche a fait, et fait toujours, l'objet de nombreux travaux [***23***], [***24***], [***25***], [***26***], [***27***], [***28***]. Il en ressort plusieurs manières de traiter le problème d'annotation automatique. Cependant, la plupart de ces méthodes nécessitent un échantillon d'apprentissage contenant des images annotées. Les annotations de nouvelles images seront apprises à partir de cet échantillon.

Grâce à des méthodes d'apprentissage automatique et à partir d'exemples d'images annotées, la plupart des techniques d'annotation automatique visent à apprendre des relations entre mots clés et caractéristiques visuelles. Les relations apprises sont ensuite utilisées afin d'attribuer des mots-clés à des images non annotées. Les méthodes d'annotation automatique semblent donc particulièrement utiles pour compléter les annotations existantes dans des bases d'images déjà partiellement annotées manuellement. Elles peuvent être divisées en deux catégories : les approches basées sur les caractéristiques locales des régions d'images [***9***], [***29***], [***30***], [***31***] et celles basées sur les caractéristiques globales d'images [***32***], [***33***]. Le premier type se fait en 3 étapes principales qui sont la segmentation d'images, l'extraction des caractéristiques des régions et enfin la génération du modèle d'annotation. On peut utiliser les algorithmes de segmentation ou tout simplement la partition de l'image en un ensemble fixe de régions similaires. Ces régions sont ensuite souvent décrites en utilisant les descripteurs de couleur, de texture, de forme, etc.… La dernière étape consiste à cartographier les régions et appliquer le modèle d'annotation déjà généré et entrainé lors de la phase d'apprentissage. L'annotation globale quant à elle contient une première étape d'extraction de caractéristiques globales de l'image et une deuxième étape de génération du modèle d'annotation. Ainsi un système d'annotation automatique d'images peut être mis en œuvre en combinant une méthode d'extraction de caractéristiques globales ou régionales et une technique d'apprentissage souvent utilisée pour déterminer la probabilité du mot clé attribué à l'image sur la base de ces caractéristiques.

Contrairement aux caractéristiques visuelles de bas niveau, le langage humain a été créé pour s'exprimer et enregistrer la connaissance humaine. Par conséquent, l'information textuelle exprimée en langage naturel se prête à la caractérisation de la sémantique dans les images. Compte tenu du manque de fiabilité du texte avoisinant ou entourant les images sur Internet et l'infaisabilité d'annoter manuellement les bases de données d'images à grande échelle, l'annotation automatique d'images, qui est destinée à faciliter la recherche et la navigation basé sur la sémantique d'images, a été étudiée pendant des années. La méthode d'annotation automatique d'images est basée essentiellement sur la reconnaissance visuelle de la machine, dont le but est de classer automatiquement les informations visuelles d'entrée en plusieurs catégories sémantiques prédéfinies.

Après avoir exposé les tâches pour l'annotation automatique d'images, un examen approfondi des approches existantes est prévu dans ce qui suit. En tant que problème de vision par machine, les méthodes d'annotations d'images peuvent être classées en fonction de nombreux aspects, tels que l'extraction de caractéristiques, la modélisation de caractéristiques, les tâches spécifiques à résoudre, la méthode d'apprentissage utilisée, etc.…

L'annotation automatique d'images a été introduite au début des années 2000, et les premiers efforts se sont concentrés sur les méthodes d'apprentissage statistique car ils fournissent des outils puissants et efficaces pour établir des associations entre les caractéristiques visuelles des images et les concepts sémantiques [***29***], [***31***], [***34***], [***35***]. Zhang et al. a proposé une étude récente sur les techniques d'annotation automatique d'images [***36***].

Les premiers travaux et efforts visant à réduire le fossé sémantique, tels que [***37***] et [***38***], ont mis l'accent sur la fourniture de mécanismes et méthodes pour cartographier les caractéristiques de bas niveau (comme la couleur, la texture, la forme et les points d'intérêt) en utilisant les caractéristiques globales. Par conséquent, ils ne prouvent leur efficacité qu'à l'égard seulement des catégories sémantiques générales (scène intérieur par rapport à scène extérieure et ville contre scène naturelle). Des approches plus avancées, comme [***39***] et [***40***], sont capables de manipuler le classement en catégories des scènes plus spécifiques. En outre, il y'a aussi des approches qui reconnaissent les événements qui se déroulent derrière la scène visuelle quand une image est prise [***41***]. Tout en utilisant des descripteurs visuels globaux et ou locaux, les approches proposées dans [***24***], [***26***], [***30***], [***31***], [***35***], [***42***], consistent à associer des mots-clés avec les objets d'une image entière en gardant les mots sur une liste classée en fonction de leurs probabilités a posteriori compte tenu de l'information visuelle.

L'emplacement et la surface des objets ne sont généralement pas pris en compte. Une autre catégorie d'annotations basée sur l'approche par région est la tâche visant à générer la correspondance la plus spécifique entre les mots et les structures d'images. Des exemples de cette catégorie comprennent [***43***], [***44***] et [***45***]. Les modèles à variables latentes proposées dans [***29***] et [***46***] sont capables de traiter l'association des mots avec les images et leurs régions.

Les approches proposées dans [***47***], [***48***], [***49***], [***50***], [***51***] qui sont basées sur la détection d'objets ou de concepts, visent à construire un jeu de concept spécifique. Elles permettent d'identifier et de localiser les concepts reconnus dans les images traitées en prédisant l'étiquette de chaque concept détecté à partir du vocabulaire d'annotation. Dans [***52***], un système de détection d'objet à base de mélanges de modèles de parties déformables multi échelle est proposé. Ces approches ont révélé une bonne précision pour la détection d'objets et concepts spécifiques, mais ils sont coûteux en calcul. Pour de plus amples informations sur les approches de détection d'objets, le lecteur peut se référer à [***53***].

Les approches basées sur la classification des images permettent de prédire la présence ou l'absence d'un ensemble de concepts du vocabulaire d'annotation. Ces approches nécessitent habituellement, soit une phase de segmentation suivi d'une autre pour l'extraction de caractéristiques de région de l'image, soit directement la phase d'extraction de caractéristiques globales et enfin une dernière phase de classification.

Quelques approches ont abordé le problème de l'annotation sémantique en utilisant des techniques d'apprentissage supervisé. La formulation de l'annotation automatique d'images comme un problème d'apprentissage supervisé a été proposée dans [***35***], [***54***]. L'approche consiste d'abord à effectuer une phase d'apprentissage. Par conséquent, les classificateurs bayésiens [***38***], les séparateurs à vastes marges (Support Vector Machines) [***55***], l'apprentissage multi-instance [***35***], [***56***], [***57***], des modèles statistiques [***58***], [***59***], k-NN [***60***], et les réseaux de neurones artificiels [***61***] sont souvent utilisés pour apprendre les concepts de haut niveau (contenu sémantique) à partir de caractéristiques de bas niveau (contenu visuel). Ce résultat est obtenu en collectant un ensemble d'images d'apprentissage pour le concept d'intérêt, et en utilisant un classificateur binaire qui est entrainé pour détecter ce concept. Le processus est répété pour chaque concept du vocabulaire d'annotation.

D'autres approches ont tenté de résoudre le problème de façon plus générale grâce à l'apprentissage non supervisé. Habituellement, ces approches utilisent des modèles probabilistes pour expliquer la cooccurrence entre les caractéristiques de l'image et des étiquettes sémantiques. Les auteurs dans [***29***], [***62***], [***63***] ont proposé l'utilisation du modèle d'allocation latente de Dirichlet (Latent Dirichlet Allocation: LDA) et des modèles graphiques pour apprendre un modèle de distribution conjointe d'un ensemble de mots-clés et les régions de l'image. Dans le même contexte, [***31***], [***64***] ont proposé l'utilisation de l'analyse sémantique latente (Latent Semantic Analysis : LSA) et analyse sémantique probabiliste latente (Probabilistic Latent Semantic Analysis : PLSA) pour l'annotation d'images.

Quelques travaux récents de l'annotation d'images ont proposé l'utilisation de l'apprentissage semi-supervisé pour tirer des avantages à la fois, des images étiquetées et non étiquetées (ou faiblement étiquetées). En effet, en raison de l'absence de données d'entraînement, l'apprentissage semi-supervisé est une bonne alternative pour annoter automatiquement les bases de données d'images à grande échelle avec des concepts sémantiques. Dans l'apprentissage semi-supervisé, un certain nombre d'exemples étiquetés sont habituellement exigés pour l'entraînement d'un prédicateur initial faiblement utile qui est à son tour utilisée pour exploiter les exemples non étiquetés [***65***]. En plus des travaux proposés dans [***66***], [***67***], [***68***], [***69***], Zhu et al. [***70***] a proposé une bonne introduction à l'apprentissage semi-supervisé (Semi Supervised Learning : SSL).

Toutes ces catégories ne sont pas optimales de manière générale, mais chacune d'entre elles peut être adaptée à un problème d'annotation d'images donné.

Comme indiqué dans [***71***], pour faire face à un grand espace sémantique composé d'un grand nombre de catégories de concepts, le processus de classification peut être amélioré par l'utilisation des hiérarchies sémantiques ou visuelles. De nombreuses approches récentes ont proposé d'utiliser des structures hiérarchiques afin de résoudre le problème de l'évolutivité dimensionnelle à grande échelle de l'annotation automatique d'images. En outre, ces hiérarchies sont porteuses de connaissances visuelles ou conceptuelles. Cette hypothèse a motivé plusieurs travaux récents qui proposent des approches pouvant être classées en méthodes descendantes (top-down) ou en méthodes ascendantes (bottom-up) selon la façon dont la hiérarchie est construite. Dans l'approche descendante, la hiérarchie des classes est construite par le partitionnement récursif de l'ensemble des classes [***55***], [***72***], [***73***], [***74***]. Dans

l'approche ascendante, la hiérarchie des classes est construite en regroupant les classes [***23***], [***75***], [***76***], [***77***], [***78***]. Ces hiérarchies sont ensuite combinées avec un ensemble de classificateurs binaires en vue de réduire la complexité du problème de classification. Récemment, deux directions principales ont été explorées pour calculer une fonction de décision pour la classification d'images hiérarchique, soit en utilisant des graphes de décision dirigés acycliques (Decision Directed Acyclic Graphs : DDAG) [***72***], [***74***], soit en utilisant des arbres de décision binaires hiérarchiques (Binary Hierarchical Decision Trees : BHDT) [***55***], [***73***], [***75***]. Les auteurs dans [***77***], [***78***], [***79***] ont proposé d'utiliser des modèles à base d'arbres n-aires (N-Ary Trees) pour résoudre le problème de la classification d'images à grande échelle. Des approches alternatives ont émergé récemment et ont proposé l'utilisation de relations sémantiques entre les concepts pour la construction ou l'utilisation des hiérarchies [***23***], [***80***], [***81***], [***82***], [***83***].

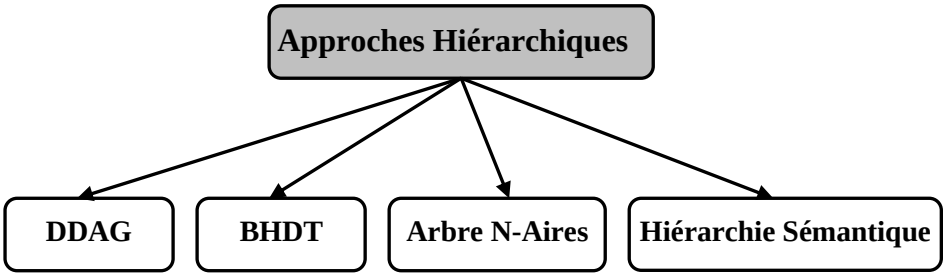


Figure II-6 : Stratégies d'annotation automatique d'images par classification hiérarchique.

La Figure II-6 illustre les différentes stratégies de l'annotation automatique d'images par classification hiérarchique décrites ci-dessus et le Tableau II-1 donne quelques informations sur leur complexité en fonction du nombre de classes N.

Tableau II-1 : Complexité de quelques stratégies d'annotation automatique d'images par classification hiérarchique.

Stratégies de l'annotation	**Phase d'apprentissage**	**Phase d'annotation**
DDAG	$O(N^2)$	O(N)
BHDT	O(N)	O(logN)
Arbre N-Aires	O(N)	O(logN)
Hiérarchie Sémantique	O(N)	O(logN)

Toutes les approches précédentes n'utilisent que les informations et les caractéristiques de bas niveau d'images pour le choix des mots clés convenables. Ceci s'avère être insuffisant pour réduire le problème du gap sémantique.

Les objets dans le monde réel sont toujours vus dans un contexte spécifique. La représentation de ce contexte est essentielle pour l'analyse et la compréhension des images. La connaissance contextuelle peut provenir de multiples sources d'informations [***84***]. Compte tenu d'un contexte particulier, ce type de connaissance peut faciliter le raisonnement sur les données afin d'améliorer l'annotation d'images. [***85***], [***86***]. Des travaux récents en vision par ordinateur ont également souligné l'importance de l'information contextuelle pour améliorer la reconnaissance d'objets dans les images du monde réel [***53***], [***87***]. Par conséquent, il est primordial de faire un usage efficace des connaissances contextuelles pour réduire le fossé sémantique et améliorer la précision de l'annotation automatique d'images.

La première tentative notable de l'utilisation des informations contextuelles pour l'annotation d'images a été proposée par [***34***]. Les auteurs ont proposé un modèle génératif statistique qui examine la probabilité d'associer des mots avec des zones d'image. Ils ont utilisé le contexte visuel environnant en calculant la probabilité jointe des caractéristiques des différentes régions de l'image. Par la suite, plusieurs approches ont proposé de fusionner et combiner des informations recueillies auprès des différentes modalités d'images afin d'améliorer la précision de l'annotation d'images. Les informations fusionnées proviennent généralement de plusieurs modalités multimédia, à savoir la fusion de l'information visuelle et textuelle [***88***], [***89***], la fusion de données textuelles et les tags des utilisateurs [***65***], [***90***], [***91***], la fusion de plusieurs images [***92***]. Clinchant et al. [***93***] a proposé trois types de techniques de fusion de l'information: la fusion précoce [***34***], [***88***], [***94***], La fusion tardive [***95***], [***96***], [***97***] et la fusion trans-média [***30***], [***98***], [***99***].

Les approches d'annotation d'images basées sur la fusion de l'information sont motivées par l'utilisation des corrélations entre les différentes modalités multimédia afin de réduire le problème de l'écart sémantique. En effet, la fusion des informations provenant de sources multiples peut réduire considérablement l'incertitude de l'annotation d'images. Cependant, ces approches ne fournissent pas nécessairement une solution fiable pour résoudre le problème de l'écart sémantique dans son intégralité. En fait, ces approches ne permettent pas d'établir des liens explicites entre les caractéristiques de l'image et la sémantique de l'image. Un enjeu important pour résoudre le problème du fossé sémantique est de faire usage

de méthodes explicites et formelles pour représenter les connaissances contextuelles. Cela permettra de prendre en compte le contexte général et spécifique de l'image, et permettra d'améliorer l'annotation d'images. L'utilisation des ontologies sémantiques, semble donc être essentielle pour s'attaquer au problème de l'écart sémantique d'une manière efficace.

Le concept de l'ontologie a émergé au début des années 1990 dans l'intelligence artificielle et les communautés de l'ingénierie des connaissances. Aujourd'hui, les ontologies sont devenues le nouveau standard pour la représentation des connaissances [***100***], [***101***]. Ces approches sont aussi utilisées pour aborder le problème de l'annotation d'images et d'interprétation [***85***], [***102***], [***103***], [***104***], [***105***], [***106***], [***107***], [***108***], [***109***]. Bien que ces tentatives intéressantes aient récemment émergé, les approches axées sur l'ontologie de l'annotation sémantique d'images sont encore à leurs débuts, et des efforts importants doivent être faits davantage afin de parvenir aux systèmes efficaces pour l'annotation d'images. En particulier, la construction automatique de ces ontologies est rarement abordée.

Pour coclure, l'annotation automatique d'images est toujours un défi important malgré plus d'une décennie de recherches. En effet, de nombreux travaux récents ont abordé cette question et ont proposé de nouvelles approches dans le but de réduire le problème de l'écart sémantique. Des solutions significatives ont été proposées pour réduire en partie ce problème de lacune sémantique. Cependant, il semble que ce n'est pas suffisant pour résoudre le problème d'annotation d'images surtout lorsqu'il s'agit de grandes bases de données d'images (grandes en termes d'images et nombre de concepts).

II.4.3 Annotation semi automatique

L'annotation semi automatique d'images est un mélange hybride entre l'annotation automatique et l'annotation manuelle. Elle est souvent proposée pour améliorer la faible précision de l'annotation automatique [***110***], [***111***]. Elle contient souvent deux parties. Tout d'abord, les images sont annotés par une méthode d'annotation automatique, puis les annotations sont mises à jour ou validées soit manuellement soit en utilisant les techniques de retour de pertinence lors du processus de recherche. Lors du processus de recherche d'images par le contenu et ou par mots clés, l'utilisateur fournit les informations concernant les images qu'il juge pertinentes ou non pertinentes. L'annotation des images est alors mise à jour en utilisant ces informations. Cependant, en plus de la tâche un peu lourde, cette stratégie d'annotation repose grandement sur les performances de recherche d'images par le contenu (CBIR). Ce qui conduit parfois à des annotations erronées et des images non considérées en

raison de l'imperfection du processus de recherches d'images qui dépend lui-même de l'indexation effectuée par l'annotation d'images.

II.5 Conclusion

Dans ce chapitre, nous avons surtout examiné le fond et le contexte de cette thèse. Nous avons présenté et définit les différents types d'annotations d'images. Nous avons aussi présenté la problématique de l'annotation d'images en relation avec l'indexation et la recherche d'images dans les bases de données. Un bref état de l'art de l'annotation automatique d'images a été aussi exposé pour avoir une idée sur l'actualité de la recherche dans ce domaine intéressant pour la navigation et l'organisation des bases de données multimédia. On a conclut enfin que l'annotation d'images est très importante pour les moteurs de recherches multimédia de telle sorte qu'elle permet de réduire le problème du gap sémantique persistant entre la perception humaine et la représentation bas niveau d'images. La structure générale d'un système d'annotation automatique d'images adopté ici ainsi que la manière d'évaluation de ses performances seront présentés brièvement dans le chapitre suivant.

Chapitre III STRUCTURE GENERALE DES SYSTEMES D'ANNOTATION AUTOMATIQUE D'IMAGES

« Il était instructif de vérifier là à quel point la fonction de l'artiste consiste, autant qu'à créer des images, à les nommer. »

[Jean Dubuffet]

Contenu du Chapitre

III.1 Introduction

Dans ce chapitre, nous allons présenter la structure générale des systèmes d'annotation automatique d'images. Les différentes parties qui composent ces systèmes seront présentées. Enfin nous allons présenter quelques méthodes d'évaluation de performances des systèmes d'annotation du contenu visuel d'images dont certaines sont empruntées aux systèmes de recherche d'images par contenu.

III.2 Structure générale des systèmes d'annotation automatique d'images

L'annotation automatique d'images peut être définie comme le processus d'attribution ou d'affectation automatique d'un ou plusieurs mots-clés à une image. Elle peut être alors considérée comme la classification automatique d'images par étiquetage en un certain nombre de classes ou catégories prédéfinies où les classes sont assignées et affectées aux mots-clés ou aux étiquettes qui peuvent décrire le contenu conceptuel d'images dans cette catégorie. Par conséquent, le problème de l'annotation automatique d'images peut être considéré comme celui de la catégorisation ou la classification automatique d'images.

Les techniques d'annotation automatiques d'images tentent d'explorer les caractéristiques visuelles d'images qui décrivent le contenu des images et de les associer au contenu sémantique de l'image. Il s'agit d'une technologie efficace pour améliorer l'indexation et la recherche de l'image dans le grand volume d'informations disponibles sur les médias. D'après l'état de l'art présenté dans le chapitre précédent, les algorithmes et les systèmes utilisés pour l'annotation d'images comportent généralement les tâches suivantes [***112***]:

- Le prétraitement et la segmentation d'images;
- L'extraction des caractéristiques d'images;
- L'annotation par classification d'images.

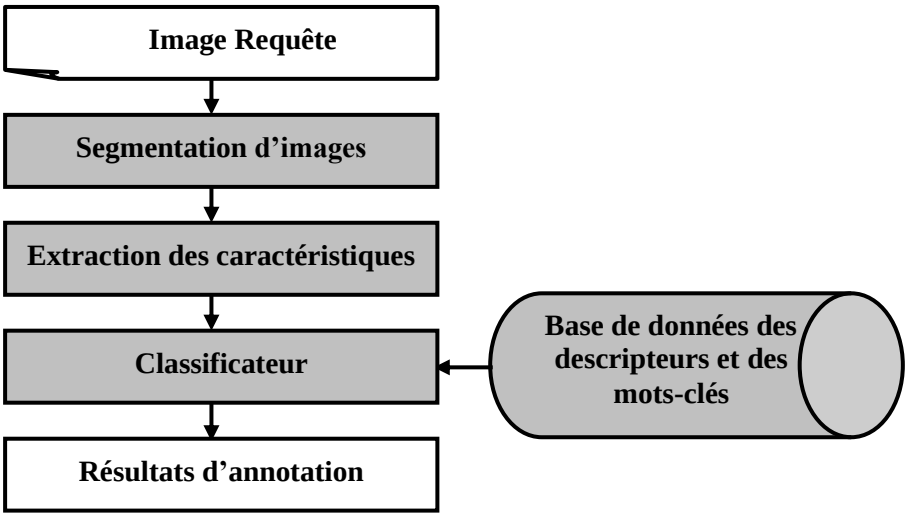


Figure III-1 : Schéma bloc du système d'annotation automatique d'images.

La Figure III-1 montre le schéma bloc du système d'annotation automatique d'images basé sur les tâches précédentes. Le système dispose d'une base de données de référence qui contient des mots-clés et des caractéristiques descripteurs d'images qui sont déjà annotées manuellement par des experts. Cette base de données est utilisée pour la modélisation et l'entrainement du classificateur afin de choisir les mots-clés appropriés. Pour atteindre cet objectif, en utilisant une méthode de segmentation, l'image d'entrée est segmentée, d'une part, en régions qui représentent des objets dans l'image. Les vecteurs caractéristiques de chaque région sont, d'autre part, calculés et extraits de l'image. Ces vecteurs caractéristiques sont finalement fournits à l'entrée du classificateur afin de décider et choisir les mots-clés appropriés pour les tâches d'annotation. Par conséquent, le contenu de l'image est annoté par des mots clés de la base de référence.

Bien que les représentations globales et par région d'images sont utilisées dans les techniques existantes de recherche d'images, la tendance pour l'annotation d'images est à l'utilisation des représentations à base de régions. L'extraction de caractéristiques à base de régions a besoin de la segmentation d'images alors que les caractéristiques globales sont calculées directement à partir de l'image entière. Dans la suite, nous passerons d'abord en revue les algorithmes de segmentation d'images couramment utilisés dans les techniques de l'annotation automatique d'images. Ensuite, diverses techniques d'extraction de caractéristiques seront survolées et examinées. Enfin, plusieurs classificateurs qui sont utilisés pour l'attribution de mots clé seront présentés.

III.2.1 Acquisition et prétraitement d'images

L'acquisition permet la conversion de l'objet en image numérique. Cette étape délicate dépend fortement du choix et du paramétrage du matériel à utiliser (Appareil photo, Scanner) ainsi que du format de stockage des images. La technicité des matériels d'acquisition (Appareil photo) a remarquablement progressé ces dernières années. La numérisation des documents est effectuée par balayage optique à l'aide d'un scanner. Le résultat est rangé dans un fichier de points, appelés pixels, dont la taille dépend de la résolution et de l'espace de couleur utilisé pour la représentation d'images.

Le prétraitement consiste à préparer les données issues du capteur d'images pour permettre davantage d'efficacité et de précision dans les phases de traitements ultérieures. Il s'agit essentiellement de réduire le bruit superposé aux données et essayer de ne garder que l'information significative de la forme représentée. Le bruit est généralement dû aux

conditions d'acquisition (éclairage, résolution du capteur, etc.). Parfois, il y'aura aussi un besoin pour le changement de la représentation de l'image d'un espace de couleur à un autre plus convenable pour un traitement particulier.

III.2.2 Segmentation automatique d'images

Le vecteur de caractéristiques extraites à partir de l'image entière perd l'information locale. Par conséquent, il est nécessaire de segmenter une image en régions ou objets d'intérêt et l'utilisation des caractéristiques locales. La segmentation d'images est le processus de partitionnement d'une image numérique en plusieurs régions représentants les objets contenu dans cette image. Elle est très importante dans de nombreuses applications de traitement d'images. Jusqu'à présent, les efforts et les tentatives sont encore entrain d'être déployés pour améliorer les techniques de segmentation. Avec l'amélioration des capacités de traitement informatique, il existe plusieurs techniques possibles de segmentation d'une image: segmentation par seuillage, segmentation par croissance de région, segmentation par la méthode des contours actifs, segmentation par la méthode des k-moyennes, etc... Dans cette thèse, nous avons utilisé la méthode de segmentation par croissance de régions ainsi que la méthode des k-moyennes [***113***], [***114***], [***115***]. La segmentation d'images est une tâhe complexe et un vaste sujet de recherche. Les performances de segmentation dépendent généralement des applications.

III.2.3 Extraction et calcul d'attributs d'images

Après avoir divisé l'image originale en plusieurs régions distinctes qui correspondent à des objets dans une scène, le vecteur de caractéristique peut être extrait à partir de chaque région, et peut être considérée comme une représentation d'un objet dans l'image entière. La tâche d'extraction de caractéristiques transforme attentivement le contenu riche et grand des données d'entrée des images en un ensemble réduit de caractéristiques afin de diminuer le temps de traitement. Il améliore non seulement la précision de la recherche et l'annotation, mais aussi la vitesse d'annotation [***116***].

La méthode d'extraction de caractéristiques d'une image doit tenir compte de la représentation du contenu de l'image dans des situations de transformations géométriques telle que: la translation, la rotation et le changement d'échelle. C'est la raison qui justifie l'utilisation limitée de certaines méthodes d'extraction des caractéristiques de l'image segmentée. La texture, les histogrammes de couleurs et les moments invariants ainsi que les

descripteurs GIST sont utilisés pour extraire les caractéristiques représentant le contenu visuel des images [***113***], [***114***], [***115***], [***117***], [***118***].

III.2.4 Annotation par classification automatique d'images

Comme cité précédemment, le problème de l'annotation automatique d'images peut être considéré comme la catégorisation ou la classification automatique d'images. Ainsi, après avoir extrait les descripteurs d'images, un classificateur approprié peut être choisi.

Il existe plusieurs types de classificateurs. Certains de ces classificateurs sont basés sur l'apprentissage supervisé, qui nécessitent une phase d'entraînement et un apprentissage intensif des paramètres du classificateur. Ils sont également connus comme classificateurs paramétriques. D'autres classificateurs fondent leur décision de classement directement sur les données, et ne nécessitent aucun entrainement et apprentissage de paramètres. Ces méthodes sont également connues sous le nom de classificateur non paramétrique. Le classificateur non paramétrique le plus courant, basé sur l'estimation de la mesure de similarité entre les caractéristiques (distance euclidienne par exemple), est le classificateur appelé le plus proche voisin.

Le classificateur permet ainsi de décider et choisir les mots-clés appropriés pour les tâches d'annotation de chaque région de l'image selon les caractéristiques qui lui sont fournies, en utilisant les paramètres d'apprentissages pour lesquels il est entrainé. Un certain nombre de classificateurs sont utilisées tels que les k plus proches voisins, les réseaux de neurones, les Séparateurs à Vaste Marge (SVM) et les réseaux bayésiens [***113***], [***114***], [***115***]. Sur la base de leur capacité à détecter des relations non linéaires complexes entre les variables dépendantes et indépendantes, chaque classificateur se trouve adapté pour classer un type particulier de vecteurs de caractéristiques.

III.3 Performance du système d'annotation automatique d'images

Une fois les images de test sont annotées par le système d'annotation automatique d'images, la qualité du système d'annotation doit être évaluée pour pouvoir effectuer la comparaison de performances entre les différents systèmes existants. Un certain nombre de mesures d'évaluation ont été utilisées par les chercheurs, dont certains sont présentés dans ce qui suit.

III.3.1 Taux d'annotation

Le taux d'annotation E d'un système d'annotation automatique d'images est donné par le rapport entre le nombre de résultats positifs r (résultats corrects) sur le nombre de tous les résultats n (résultats utilisés pour le test du système). Il est donné par :

$$E = \frac{r}{n} \qquad \textbf{(III.1)}$$

III.3.2 Rappel et précision

Le rappel et la précision sont deux mesures classiques en recherche d'information qui peuvent être empruntées et utilisés en annotation automatique d'images. Le taux de précision P (Precision) est le nombre de résultats corrects r, divisé par le nombre de tous les résultats retournés (la somme du nombre de résultats corrects r et le nombre de résultats incorrects w) et le taux de rappel R (Recall) est le nombre de résultats corrects r, divisé par le nombre de résultats qui auraient dû être retournés n (pertinents).

Soient n le nombre de mots-clés pertinents, r le nombre de mots-clés pertinents retrouvés et w le nombre de mots-clés non-pertinents retrouvés. Le rappel R et la précision P sont définis par :

$$R = \frac{r}{n} \qquad et \qquad P = \frac{r}{r+w} \qquad \textbf{(III.2)}$$

III.3.3 Mesure F1

La précision de l'annotation d'images peut être évaluée aussi par la mesure F1 qui est une valeur intégrée du débit de précision et le taux de rappel. La mesure F1 peut être interprétée comme une moyenne pondérée des taux de précision et de rappel, où le score F1 atteint sa meilleure valeur à 1 et sa pire valeur à 0. La mesure F1 est définit par [***119***]:

$$F_1 = \frac{2PR}{P+R} \qquad \textbf{(III.3)}$$

Pour mesurer les performances d'un modèle d'annotation automatique d'images, il est important de prendre aussi en compte d'autres facteurs tels que :

- le nombre de paramètres,

- la capacité du modèle à prédire sur de nouvelles données,
- la capacité du modèle à obtenir de bons résultats lorsque le nombre de mots du vocabulaire, de dimensions visuelles et/ou de données augmentent.

La valeur des scores obtenus dépend du nombre de mots dans le vocabulaire, ainsi que de la difficulté de la tâche, et des données.

Toutes ces méthodes de mesure de performance d'un système d'annotation automatique d'images nécessitent la connaissance des annotations correctes des images. C'est-à-dire qu'un lourd travail d'annotation manuelle de la base de test est nécessaire pour pouvoir comparer les annotations estimées aux annotations réelles. De plus, la probabilité pour que deux personnes annotent une région d'images avec le même mot est faible. Ceci complique la tâche d'évaluation des systèmes d'annotations automatiques d'images.

Enfin, la diversité des méthodes d'évaluation rend la comparaison des systèmes d'annotations existants très difficile. Il existe donc des campagnes d'évaluation et des compétitions dédiées aux problèmes d'annotation [***120***], [***121***], qui proposent des bases d'images standards permettant la comparaison des méthodes.

III.4 Conclusion

Dans ce chapitre, nous avons présenté brièvement la structure générale des systèmes d'annotation automatique d'images. Un bref aperçu des différentes parties des systèmes d'annotation automatique d'images a été aussi introduit et fourni. En fin, nous avons introduit plusieurs mesures d'évaluation des performances des systèmes automatiques d'annotations d'images, dont la précision et le rappel sont les plus populaires. La conception et la réalisation du système d'annotation automatique d'image adopté seront présentées en détail dans les chapitres de la deuxième partie du présent document.

Partie 2 : CONCEPTION & REALISATION D'UN SYSTEME D'ANNOTATION AUTOMATIQUE D'IMAGES

Chapitre IV CONCEPTION DU SYSTEME D'ANNOTATION AUTOMATIQUE D'IMAGES

« Le langage est la peinture de nos idées, qui à leur tour sont des images plus ou moins étendues de quelques parties de la nature. »

[Antoine de Rivarol]

Contenu du Chapitre

IV.1 Introduction

L'objectif de ce chapitre est de concevoir la structure du système d'annotation automatique d'images à réaliser. Le fondement théorique et le formalisme mathématique des différentes parties du système d'annotation automatique d'images conçu et réalisé dans le cadre de cette thèse seront fournis et présentés en détail. Nous allons définir et détailler les principales parties qui composent le système d'annotation automatique d'images proposé, à

savoir : la segmentation d'images, l'extraction d'attributs d'images et l'annotation par classification d'images. Quelques méthodes de segmentation d'images sont présentées comme étant la première phase du système d'annotation automatique d'images. Pour la deuxième phase, les méthodes d'extractions et d'exploitations des paramètres descriptifs du contenu visuel d'images sont définies et étudiées. Pour la dernière phase du système, les techniques de classifications utilisées comme approche d'annotation automatique d'images sont aussi définies et décrites en détail.

IV.2 Structure du système d'annotation automatique d'images

D'après l'état de l'art présenté dans le chapitre précédent, pour annoter le contenu visuel d'une image, il faut d'abord la segmenter en régions homogènes qui sont supposées représenter les objets contenus dans cette image. Il faut ensuite extraire les paramètres descripteurs de chaque région afin d'établir une relation avec les mots visuels ou les mots-clés par un classificateur entrainé pour ce but. La structure générale d'un tel système d'annotation automatique d'images est représentée par la Figure IV-1. Cette structure se compose principalement des parties principales suivantes :

- Segmentation automatique d'images ;
- Extraction des descripteurs d'images ;
- Classification et annotation d'images.

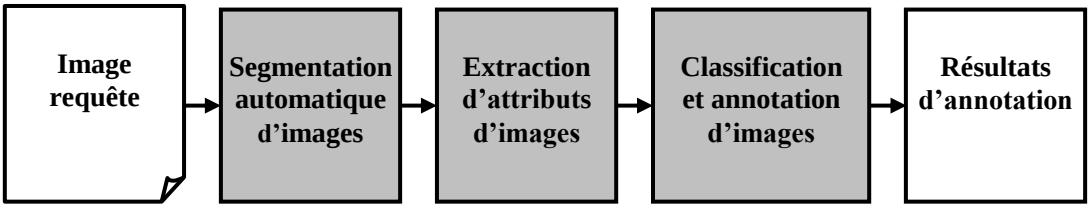


Figure IV-1 : Structure du système d'annotation automatique d'images

Les différentes parties du système d'annotation automatique d'images seront étudiées et présentées dans les sections suivantes.

IV.3 Segmentation automatique d'images

Habituellement, le vecteur des caractéristiques extraites de l'image entière perd l'information locale. Par conséquent, il est nécessaire de segmenter une image en régions d'intérêt ou objets afin d'utiliser les caractéristiques locales.

IV.3.1 Notion de segmentation d'images

La segmentation de l'image est une méthode qui localise et extrait un objet à partir d'une image ou divise l'image en plusieurs régions.

La segmentation cherche à partitionner une image en sous-ensembles de régions disjoints et connexes R_i, tel que chaque région soit homogène et que l'union de deux régions adjacentes ne le soit pas. Une segmentation est un partitionnement de l'image en régions respectant cette définition pour un prédicat donné $P(:)$ souvent lié à un critère d'homogénéité [***122***] [***123***] [***124***]. D'un point de vue mathématique, Zucker [***125***] définit une segmentation $S = \{R_1, R_2, ..., R_n\}$ de l'ensemble des pixels d'une image comme l'ensemble des régions R_i satisfaisant et vérifiant les propriétés suivantes:

1) $\bigcup_{i=1}^{n} R_i = \text{Im}\,age$;
2) $R_i \cap R_j = \{\phi\} \quad \forall\, i \neq j$;
3) $P(R_i) = Vrai \quad \forall\, i = 1, \ldots, n$;
4) $P(R_i \cup R_j) = Faux \quad \forall i \neq j \;\; et \;\; R_i \; adjacente \; à \; R_j$;
5) $p \; connecté \; à \; q \quad \forall\, p, q \, des \; pixels \in R_i$.

La proposition $P(R_i)$ signifie l'homogénéité de la région R_i qui peut être vraie ou fausse. La première propriété signifie que tout pixel de l'image appartient à une région de l'image, tandis que la deuxième signifie qu'aucun pixel n'appartient à plus d'une région. Toute région dans une image doit vérifier le prédicat d'homogénéité tel qu'il est précisé par la 3ème propriété. La 4ème propriété précise que l'union de deux régions ne doit en aucun cas être homogène, sinon ces deux régions ne sont qu'une seule en fin de compte. La dernière propriété montre que si deux pixels ou point appartiennent à une même région R_i, ils doivent nécessairement être connectés entre eux.

Il existe plusieurs techniques possibles de segmentation d'images: le seuillage, la croissance de régions, les contours actifs, K-moyennes, etc... [***122***] [***123***] [***124***] [***125***] [***126***]. Chacune d'elles intègre d'une façon ou d'une autre un terme d'attache aux données qui est le nerf de la segmentation. Ce terme d'attache aux données permet de guider la segmentation en fonction de l'image. Les méthodes de segmentation sont aussi indénombrables que les applications visées ainsi que les images à traiter. Pour vérifier les propriétés du prédicat d'homogénéité, elles prennent en compte deux types d'information: L'information photométrique liée à la couleur, le contraste, la texture de l'image et l'information géométrique du contour. Tandis que la première information se fonde sur le niveau de gris des pixels, ou bien sur une texture locale, la seconde exprime les propriétés du contour de la région segmentée et la géométrie de l'objet à extraire. Cette dernière information permet également d'apporter une donnée supplémentaire à la segmentation lorsque l'image est fortement bruitée ou faiblement contrastée. Bien que l'association de ces deux termes contribue à l'amélioration des résultats de segmentation, elle peut parfois se montrer inefficace pour segmenter des données corrompues ou des images trop fortement bruitées.

La segmentation d'images consiste alors à attribuer un même label aux zones de l'image possédant des propriétés homogènes : en termes d'intensité, de couleur, de texture, de zones délimitées par un fort gradient d'intensité et de contour, tout en respectant les propriétés citées précédemment. L'imprécision et l'inefficacité d'une méthode de segmentation d'images dépend fortement du degré de violation de ces propriétés.

IV.3.2 Méthode par croissance de régions

Parmi les méthodes de segmentation d'images les plus utilisées, la méthode par croissance de régions (region growing en anglais) est bien adaptée en raison de sa simplicité. Elle a été utilisée avec succès à plusieurs reprises [***126***]. D'un ensemble de points initiaux, les régions sont augmentées de manière itérative en comparant tous les pixels voisins non affectés à ceux de la région. La différence entre l'intensité d'un pixel et la moyenne de la région est utilisée comme une mesure de similitude, c'est un prédicat qui contrôle l'évolution de la segmentation. Le pixel avec la plus petite différence mesurée de cette manière (inférieur au seuil 0.2) est affecté à la zone respective. Ce processus est répété jusqu'à ce que tous les pixels soient affectés à une région. En supposant que les objets sont au centre, dans la plupart des images, la segmentation par croissance de région est débutée à partir de l'angle de l'image

afin d'isoler les objets dans le centre de l'image. Ces étapes sont présentées de façon plus détaillée dans l'Algorithme IV.1.

Étant donné un ensemble de pixels d'une image $X = \{p_1, p_2, \ldots, p_n\} \in IR^d$, où chaque pixel est un véritable vecteur de dimension d=3 dans le cas d'une image couleur (d=5 si on introduit les cordonnées des pixels comme informations de cohérence spatiale ou de connexité), l'algorithme de segmentation d'images par croissance de région qui permet de segmenter l'image X en une carte de régions $S = \{R_1, R_2, \ldots, R_k\}$ est donné par :

Algorithme IV.1 : Segmentation d'images par la méthode de croissance de régions.

Algorithme : Segmentation d'images par la méthode de croissance de régions.

Entrée : L'image $X = \{p_1, p_2, \ldots, p_n\} \in IR^d$ et le Seuil ;

Sortie : Carte de régions $S = \{R_1, R_2, \ldots, R_k\}$;

Début

$j \leftarrow 1$; *//étiquette de la région.*

Tant Que (l'image n'est pas entièrement segmentée) **Faire**

1- Choisir un pixel p non étiqueté;

2- Fixer la moyenne de la région sur l'intensité du pixel p;

3- Considérer les pixels voisins non étiqueté p_i ;

Si $\left| \textit{Intensité du pixel } p_i - \textit{Moyenne de la région } R_j \right| \leq \textit{Seuil}$ | **Alors**

Ajouter le pixel p_i à la région R_j ;

Marquer l'étiquette du pixel p_i dans la carte de régions par j ;

Mettre à jour la moyenne de la région R_j ;

retour à l'étape **3**;

Sinon

$j \leftarrow j + 1$;

retour à l'étape **1** ;

Fin Si;

Fin Tant Que;

Retourner Les clusters (Régions) $S = \{R_1, R_2, \ldots, R_k\}$;

Fin.

IV.3.3 Méthode des K-Moyennes (K-Means)

La méthode des K-Moyennes (k-means en anglais) est avant tout un outil de classification classique qui permet de répartir un ensemble de données en k classes homogènes. Du fait que la plupart des images numériques vérifient localement des propriétés d'homogénéité, notamment en termes d'intensité lumineuse, l'algorithme des K-Moyennes permet donc d'apporter une solution à la segmentation d'images.

Vers les années 60, La méthode des K- Moyennes a été introduite par J. McQueen [***127***] et mise en œuvre sous sa forme actuelle par E. Forgy [***128***]. De nombreuses variantes se sont succédées depuis afin d'étendre ses capacités de classification (séparations non linéaires basées sur des méthodes à noyaux), améliorer ses performances, ou automatiser le choix du nombre de clusters. [***129***], [***130***] [***131***], [***132***], [***133***].

Dans le cadre de la classification non supervisée, on cherche généralement à partitionner l'espace en classes concentrées et isolées les unes des autres. L'algorithme K-Moyennes est un algorithme basé sur la classification non supervisée qui ne nécessite pas la présence d'une base d'apprentissage. Ce qui fait que cet algorithme permet d'organiser les pixels de l'image sous forme de classes suivant un ou plusieurs critères d'optimisation et d'homogénéité [***130***].

Étant donné un ensemble de pixels d'une image $X = \{p_1, p_2, \ldots, p_n\} \in IR^d$, où chaque pixel est un véritable vecteur de dimension d=3 dans le cas d'une image couleur (d=5 si on introduit les cordonnées des pixels comme informations de cohérence spatiale ou de connexité), l'algorithme k- Moyennes vise à diviser et classer les n pixels de l'image X en k ensembles ou régions (k ≤ n) $S = \{R_1, R_2, \ldots, R_k\}$ de manière à minimiser la variance intra-classe, qui se traduit par la minimisation de l'erreur quadratique définie par :

$$E = \sum_{i=1}^{k} \sum_{p_j \in R_i} \left\| p_j - m_i \right\|^2 = \sum_{i=1}^{k} Card(R_i) \times Var(R_i) \qquad \textbf{(IV.1)}$$

Où : p_j vecteur du pixel j ;

- $Card(R_i)$ est le nombre de pixels de la région R_i (cardinal de R_i) ;

- $m_i = \frac{\sum_{p_j \in R_i} p_j}{Card(R_i)}$ est le centre de la région R_i appelé aussi noyau ;
- $Var(R_i) = \frac{\sum_{p_j \in R_i} \|p_j - m_i\|^2}{(Card(R_i))^2}$ est la variance des pixels du cluster ou région R_i .

L'algorithme de segmentation d'images par la méthode des K-moyennes trouvera alors les k groupes de pixels qui minimisent la quantité E définie précédemment. Ce critère est appelé parfois erreur de regroupement (clustering) et dépend des centres de classes. Il affecte chaque pixel à un cluster donné de telle sorte que cette erreur soit minimisée. Ce qui revient en quelque sorte, pour chaque cluster ou région R_i, à minimiser la quantité suivante :

$$\sum_{p_j \in R_i} \|p_j - m_i\|^2 \qquad \textbf{(IV.2)}$$

Le principe de l'algorithme de minimisation de cette erreur peut se traduire par les principales étapes suivantes [***134***]:

1. Choix du nombre de clusters (nombre de noyaux).
2. Initialisation des clusters et leurs noyaux.
3. Mise à jour des clusters en optimisant l'erreur de regroupement.
4. Calcul et réévaluation des noyaux des nouveaux clusters.
5. Itérer et répéter les étapes 3 et 4 jusqu'à stabilisation des noyaux des clusters.

La présentation de ces étapes de façon un peu plus formelle est donnée par l'Algorithme IV.2 suivant :

Algorithme IV.2 : Segmentation d'images par la méthode des k-moyennes.

Algorithme : Segmentation d'images par la méthode des k-moyennes.

Entrées : L'image $X = \{p_1, p_2, \ldots, p_n\} \in IR^d$;

Sorties : Les clusters (carte de Régions) $S = \{R_1, R_2, \ldots, R_k\}$;

Début

Déterminer k selon le principe de la Figure IV-2;

// initialisation des clusters

Pour i=1 **à** k **Faire**

// Tirage aléatoire pour initialiser les noyaux (centres) des clusters.

$m_i \leftarrow p \quad / \; p \in X$;

// Attribuez au cluster R_i les pixels les plus proches de lui.

$$R_i \leftarrow \left\{ p \in X \quad / \min_{1 \le j \le k} \left\| p - m_j \right\|^2 = \left\| p - m_i \right\|^2 \right\} ;$$

Fin Pour ;

Répéter

// Initialiser la condition d'arrêt.

Test de stabilisation des noyaux $\leftarrow$ *Vrai* ;

Pour i=1 **à** k **Faire**

// Mise à jour du cluster R_i par les pixels les plus proches de lui.

$$R_i \leftarrow \left\{ p \in X \quad / \min_{1 \le j \le k} \left\| p - m_j \right\|^2 = \left\| p - m_i \right\|^2 \right\} ;$$

// si le noyau ou le centre du cluster R_i est instable

Si $\left(m_i \neq \frac{1}{Card(R_i)} \times \sum_{p \in R_i} p \right)$ **Alors**

// Mettre à jour le noyau instable

$$m_i \leftarrow \frac{1}{Card(R_i)} \times \sum_{p \in R_i} p \; ;$$

// Modifier la condition d'arrêt.

Test de stabilisation des noyaux $\leftarrow$ *Faux* ;

Fin Si ;

Fin Pour ;

Jusqu'à (*Test de stabilisation des noyaux = Vrai*) ;

Retourner Les clusters (Régions) $S = \{R_1, R_2, \ldots, R_k\}$;

Fin.

L'Algorithme IV.2 défini précédemment converge vers un minimum local de l'erreur de regroupement des clusters, qui se traduit par une partition de l'image en des régions séparées. Les propriétés de convergence de cet algorithme ont été étudiées et vérifiées par [***135***].

La qualité de la solution ainsi trouvée dépend fortement des noyaux initiaux. De plus la sensibilité de l'algorithme à l'initialisation est d'autant plus grande que la dimensionnalité des données est grande. De nombreuses stratégies ont été élaborées afin d'établir une bonne initialisation aux k-moyennes ou bien les rendre moins sensibles aux noyaux initiaux [***136***], [***137***], [***138***], [***130***], [***131***].

Généralement le choix du nombre de clusters k est fait empiriquement en sélectionnant la valeur de k qui minimise l'erreur de regroupement déjà définie un peu plus haut. Différents critères permettent d'estimer le nombre de clusters en minimisant la distance intra-classes tout en maximisant la distance interclasses [***139***]. De nombreuses stratégies permettent de déterminer le nombre de clusters pendant le déroulement de l'algorithme. Ces méthodes sont basées sur un processus itératif au cours duquel on choisit de subdiviser les clusters précédemment établis en se basant sur un critère statistique [***132***] [***133***].

Le nombre de clusters dans le cas de la segmentation d'images peut correspondre au nombre de couleurs ou au nombre d'intensités utilisées pour représenter l'image. En se basant sur ce principe, la détermination de k est faite en utilisant les histogrammes de couleurs. La Figure IV-2 représente le principe utilisé pour le choix de k.

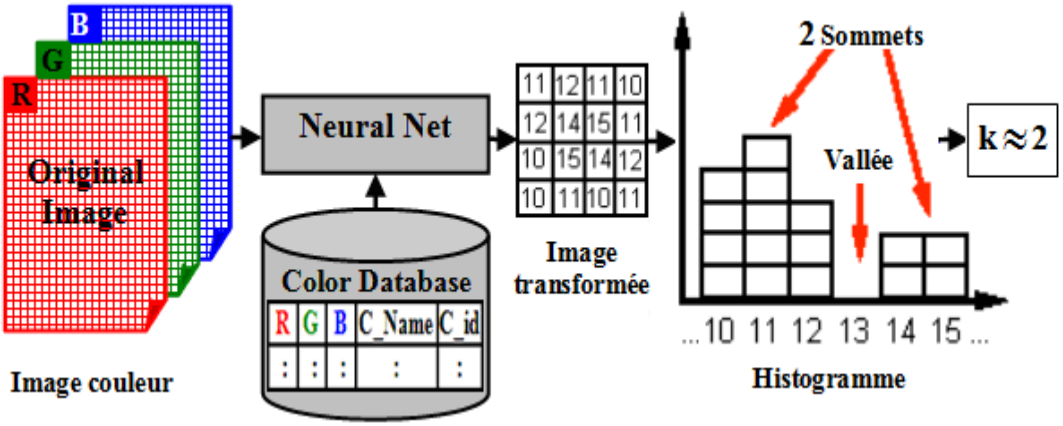


Figure IV-2 : Principe utilisé pour le choix de k à partir d'une image couleur.

Après la transformation de l'image couleur en une image simple formée par les indices des couleurs réduites, le nombre de cluster k est choisi comme étant le nombre de sommets dans l'histogramme de l'image transformée. La transformation de couleur est obtenue en utilisant les valeurs RVB (RGB) des couleurs simples du Web et un réseau neuronal spécial conçu et entraîné pour cette tâche. Cette transformation est utilisée ultérieurement pour déterminer aussi les couleurs dominantes dans une image couleur.

La méthode de classification de pixels (clustering) basée sur l'algorithme des K-moyennes constitue donc une approche naturelle pour réaliser une segmentation d'images. Néanmoins, cette classification ne tient pas compte des informations spatiales des pixels d'une image, on constate donc des problèmes de connexité aux frontières.

IV.3.4 Représentation des régions

Lors du processus de la segmentation d'images, chaque région doit être identifiée par une étiquette unique. Un plan d'étiquette est alors nécessaire afin de faciliter l'identification et le traitement des régions dans une image. Le plan d'étiquette est une image de même dimension et parallèle au plan d'images d'origine. Il indique les étiquettes correspondantes aux pixels. Il peut être représenté par une simple matrice 2-D. Ce plan d'étiquettes permettra d'isoler n'importe quelle région voulue. La Figure IV-3 représente un exemple d'image et son plan d'étiquettes de régions.

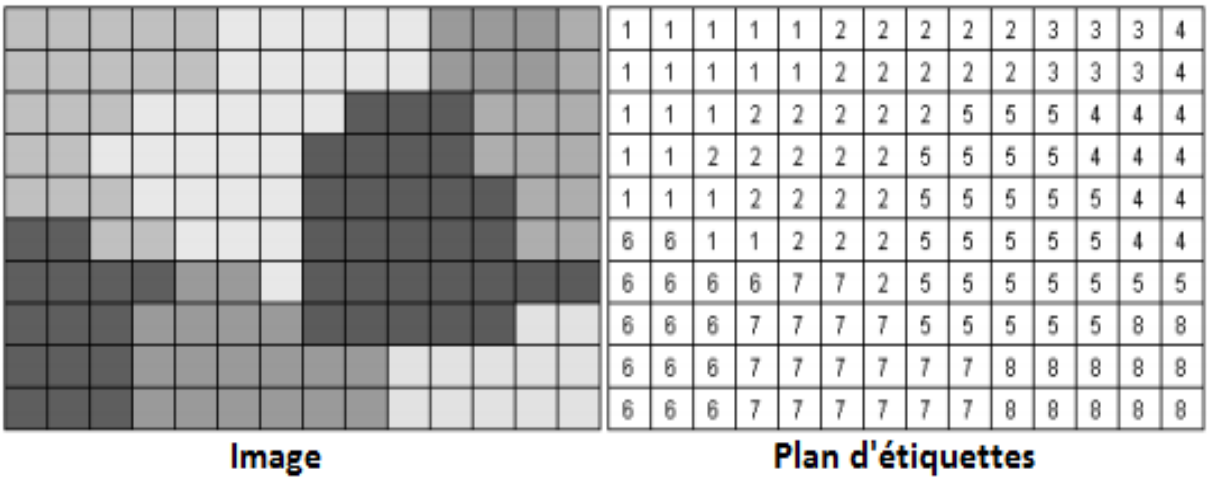


Figure IV-3 : Image et son plan d'étiquettes de régions.

Après avoir divisé l'image originale en plusieurs régions distinctes qui correspondent à des objets dans une scène, le vecteur de caractéristique doit être soigneusement extrait à partir de chaque région afin de réduire la grandeur des données d'entrée d'une image tout en préservant la représentation riche du contenu de l'image entière. Ceci fera l'objet de la section suivante.

IV.4 Extraction d'attributs d'images

Lorsque les données d'entrée à un algorithme sont trop volumineuses pour être traitées, elles sont transformées en une représentation réduite des caractéristiques définies. Transformer les données d'entrée en un ensemble réduit de caractéristiques est appelé extraction de caractéristiques de l'image. La tâche d'extraction transforme un contenu riche et divers de l'image en un ensemble de caractéristiques réduites. Afin d'effectuer cette tâche qui permet l'utilisation d'une représentation réduite à la place des données d'entrée en taille réelle, ces caractéristiques doivent être extraites rigoureusement et soigneusement [***112***].

L'extraction améliore non seulement la précision de la recherche et l'annotation d'images, mais aussi la vitesse de l'annotation. Ainsi, une base de données volumineuse d'images peut être organisée selon la règle de classification et donc, la recherche peut être effectuée en utilisant ces informations réduites [***116***].

Il existe plusieurs techniques d'extractions de ces informations réduites qui diffèrent par leur précision et leur dimension. Dans ce qui suit, nous allons présenter les techniques d'extraction d'attributs d'une image utilisées dans le système d'annotation que nous proposons. Dans le procédé d'extraction de caractéristiques, la représentation du contenu de l'image doit être considérée dans certains cas tels que: la translation, la rotation et le changement d'échelle. C'est la raison qui justifie l'utilisation d'histogrammes de couleurs, des moments invariants, des descripteurs GIST et la texture comme méthodes d'extraction de caractéristiques d'images déjà segmentées. Pour plus de précision et d'exactitude dans le système d'annotation, ils peuvent être combinés ensemble et se présenter à l'entrée du classificateur [***140***]. Cette combinaison coûte plus de temps pour l'entraînement des classificateurs en raison de la grande taille des éléments aboutis.

IV.4.1 Histogramme de couleurs

Typiquement, la couleur d'une image est représentée par un certain modèle de couleur. Il existe différents modèles de couleur pour décrire l'information de couleur. Les modèles de couleurs les plus couramment utilisés sont RVB (rouge, vert, bleu), le HSV (teinte, saturation, value) et Y, Cb, Cr (luminance et chrominance). Ainsi, la teneur de couleur est caractérisée par trois canaux à partir des modèles de couleur définis précédemment.

Une représentation du contenu d'images en couleurs se fait en utilisant un histogramme de couleurs. Statistiquement, l'histogramme désigne la probabilité conjointe des intensités des trois canaux de couleurs [***112***].

L'histogramme d'une image est la fonction qui associe, à chaque valeur d'intensité, le nombre de pixels dans l'image ayant cette valeur. Un histogramme de couleur décrit alors la répartition des couleurs dans l'ensemble ou dans une région d'intérêt de l'image. L'histogramme est invariant à la rotation, à la translation et au changement de l'échelle d'un objet. Mais l'histogramme ne contient aucune information sémantique, et deux images avec des histogrammes de couleurs similaires peuvent posséder des contenus différents.

Les histogrammes sont généralement divisés en bins dans le but de représenter grossièrement le contenu et réduire la dimensionnalité de classification ultérieure et la phase d'adaptation (correspondance). La réduction se fait simplement en effectuant une quantification plus forte de l'histogramme. Pour réduire un histogramme de 256 à k valeurs, on découpe l'histogramme en morceaux de $256/k$ valeurs. Pour une image donnée, un histogramme de couleur est défini en tant que vecteur par:

$$H = \left\{ h[i] = \underset{i \in \{1,\ldots,k\}}{Card} \left\{ p \ / \ (i-1) * E\left(\frac{256}{k}\right) \leq color(p) < i * E\left(\frac{256}{k}\right) \right\} \right\} \qquad \textbf{(IV.3)}$$

Avec:

- i représente un bin dans l'histogramme de couleurs;
- p représente un pixel de l'image ;
- h [i] est le nombre de pixels du bin i dans l'histogramme de couleurs de cette image;
- E(x) est la partie entière de x ;
- Card(x) est le nombre d'éléments dans x ;
- k est le nombre de bins dans le modèle de couleur adopté.

Pour être invariant à l'échelle des objets dans des images de tailles différentes, les histogrammes de couleurs doivent être divisés par le nombre total de pixels $M \times N$ d'une image afin d'obtenir les histogrammes de couleurs normalisés. Ainsi, pour une image f de taille M x N pixels, l'histogramme est défini par :

$$H = \left\{ h[i \in \{1,...,k\}] = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\delta(f(x,y)-C(i))}{M \times N} \;/\; (i-1)\times E\left(\frac{256}{k}\right) \le C(i) < i \times E\left(\frac{256}{k}\right) \right\} \quad \textbf{(IV.4)}$$

Dans cette équation, δ représente l'impulsion unitaire de Dirac définie par :

$$\delta(x) = \begin{cases} 1 & si \quad x = 0 \\ 0 & si \quad x \neq 0 \end{cases} \quad \textbf{(IV.5)}$$

L'algorithme permettant de calculer les histogrammes de couleurs d'une image f (Algorithme IV.3) est donné par :

Algorithme IV.3 : Calcul de l'histogramme réduit d'une image couleur.

Algorithme : calcul de l'histogramme réduit d'une image couleur.

Entrées : l'image couleur f de taille $M \times N \times 3$, Le nombre de bins k ;

Sorties : les vecteurs des histogrammes des 3 canaux de couleurs h_R , h_G et h_B ;

Début

Pour i=1 **à** k **Faire** // Initialisation

$h_R[i] = h_G[i] = h_B[i] \leftarrow 0$;

Fin Pour ;

Pour x=0 **à** M-1 **Faire**

Pour y=0 **à** N-1 **Faire**

$$h_R\left[E\left(\frac{f(x,y,1)}{k}\right)+1\right] \leftarrow h_R\left[E\left(\frac{f(x,y,1)}{k}\right)+1\right]+1;$$

$$h_G\left[E\left(\frac{f(x,y,2)}{k}\right)+1\right] \leftarrow h_G\left[E\left(\frac{f(x,y,2)}{k}\right)+1\right]+1;$$

$$h_B\left[E\left(\frac{f(x,y,3)}{k}\right)+1\right] \leftarrow h_B\left[E\left(\frac{f(x,y,3)}{k}\right)+1\right]+1;$$

Fin Pour ;

Fin Pour ;

Retourner $\frac{h_R}{M \times N}$, $\frac{h_G}{M \times N}$ et $\frac{h_B}{M \times N}$;

Fin.

Pour une image à trois canaux de couleurs, nous aurons un histogramme pour chaque canal de couleur. Un vecteur de caractéristiques peut être alors formé par la concaténation des trois histogrammes de canaux en un seul vecteur. La Figure IV-4 présente un exemple d'image et ses 3 histogrammes de couleurs.

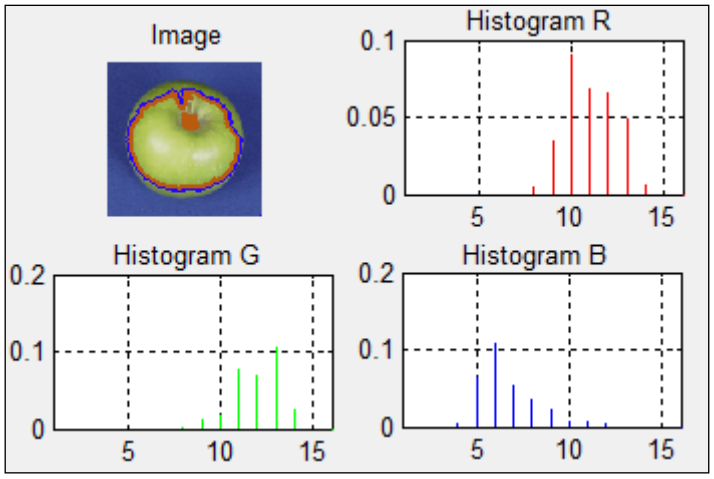


Figure IV-4 : Exemple d'histogrammes de couleurs.

En plus de leur utilisation en tant que descripteurs de caractéristiques d'images, les histogrammes de couleurs sont également utilisés pour déterminer la couleur dominante d'un objet dans une image. Comme le montre la Figure IV-5, la couleur avec un histogramme maximale est considérée comme étant la couleur dominante. Puisque l'histogramme de l'image couleur se compose de 3 canaux, l'image est transformée en un seul canal à partir de 3 canaux en utilisant une carte de transformation de couleurs et l'histogramme de l'image transformée est calculé ensuite. La carte de transformation de couleur est obtenue en utilisant les valeurs RVB (RGB) des couleurs simples du Web et un réseau neuronal spécial conçu et entraîné pour cette tâche. Après avoir calculé l'histogramme de l'image résultante de la transformation, l'histogramme maximum est déterminé. La même carte de transformation de couleur est utilisée afin de transformer la valeur maximale de l'histogramme de couleur à ses valeurs RVB correspondantes. Par conséquent, les objets de l'image peuvent être annotés également par les noms de leur couleur dominante. La couleur peut être considérée comme dominante si le rapport entre le premier maximum et le deuxième maximum est supérieur à un seuil choisi.

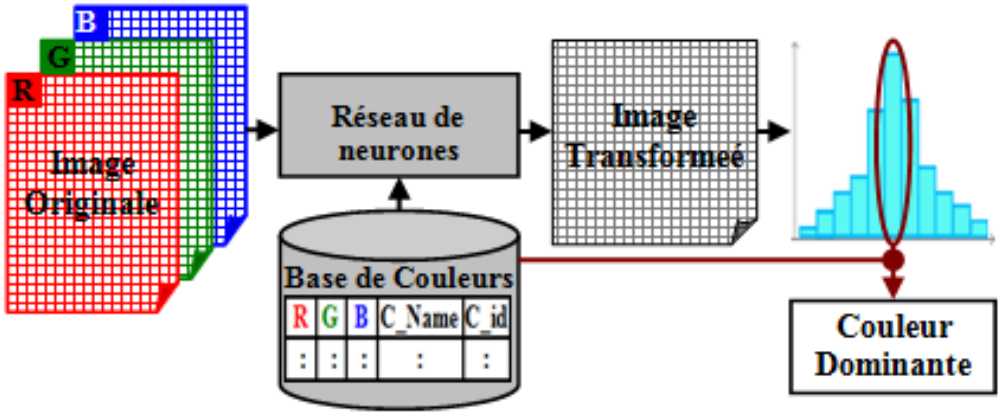


Figure IV-5 : Schéma de principe pour déterminer la couleur dominante dans l'image d'entrée.

La carte de transformation peut être obtenue en utilisant d'autres types de classificateurs autre que le réseau neuronal tels que les SVM et les réseaux bayésiens ainsi que d'autres espaces de couleurs.

IV.4.2 Moments Invariants

L'utilisation des moments pour l'analyse d'image et la reconnaissance de motif a été inspiré par Hu [***141***] et Alt [***142***]. Les moments les plus couramment utilisés sont les suivants:

- Les moments de Hu.
- Les moments de Legendre.
- Les moments de Zernike.

IV.4.2.1 Moments de Hu

Les moments de Hu sont largement utilisés dans le traitement d'images et reconnaissance de formes. Ils sont dérivés et calculés à partir des moments géométriques.

Le moment géométrique bidimensionnelle d'ordre (p + q) d'une image représentée par une fonction réelle mesurable de valeur f (x, y) dans l'intervalle $[a_1, a_2] \times [b_1, b_2]$ est définie par:

$$M_{pq} = \int_{a_1}^{a_2}\int_{b_1}^{b_2} x^p y^q f(x,y)\, dxdy \qquad \textbf{(IV.6)}$$

Où p, q=0, 1, 2, .., ∞.

Le produit de monômes $x^p y^q$ constitue la fonction de base de la définition des moments géométriques. Un ensemble de n moments se compose de tout M_{pq} tel que $p+q \le n$.

L'ordre zéro, M_{00}, des moments géométriques représente le total de la masse de l'image f(x, y). Il est définit par :

$$M_{00} = \int_{a_1}^{a_2}\int_{b_1}^{b_2} f(x,y)\, dxdy \qquad \textbf{(IV.7)}$$

Les deux premiers moments représentent le centre de masse de l'image f(x, y). Ils sont donnés par :

$$\begin{aligned} M_{10} &= \int_{a_1}^{a_2}\int_{b_1}^{b_2} x\ f(x,y)\, dxdy \\ M_{01} &= \int_{a_1}^{a_2}\int_{b_1}^{b_2} y\ f(x,y)\, dxdy \end{aligned} \qquad \textbf{(IV.8)}$$

En termes de valeurs des moments, les coordonnées du centre de masse sont :

$$\bar{x} = \frac{M_{10}}{M_{00}} \quad , \quad \bar{y} = \frac{M_{01}}{M_{00}} \qquad \textbf{(IV.9)}$$

Les moments centraux d'une image représentée par f (x, y) sont définis comme suit:

$$\alpha_{pq} = \int_{a_1}^{a_2}\int_{b_1}^{b_2} \left(x-\bar{x}\right)^p \left(y-\bar{y}\right)^q f(x,y)\, dxdy \qquad \textbf{(IV.10)}$$

Les moments centraux définis dans l'équation (IV.10) sont invariants par la translation de coordonnées. Ils peuvent être normalisés pour préserver l'invariance par changement d'échelle. Pour $p+q = 2, 3, \ldots$, les moments centraux normalisés d'une image sont donnés par:

$$\mu_{pq} = \frac{\alpha_{pq}}{\alpha_{00}^{\ \gamma}} \quad avec \quad \gamma = \frac{p+q}{2} + 1 \qquad \textbf{(IV.11)}$$

En se basant sur la théorie de l'invariance algébrique, Hu [***141***] a dérivé des combinaisons relatives et absolues des moments qui sont invariantes aux changements d'échelle, de translation et de rotation. Hu a défini les sept fonctions suivantes, calculées à partir des moments centraux normalisés de l'ordre de trois, qui sont invariantes aux changements d'échelle, de translation et de rotation:

$$\phi_1 = \mu_{20} + \mu_{02} \quad \textbf{(IV.12)}$$

$$\phi_2 = (\mu_{20} + \mu_{02})^2 + 4\mu_{11}^2 \quad \textbf{(IV.13)}$$

$$\phi_3 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2 \quad \textbf{(IV.14)}$$

$$\phi_4 = (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2 \quad \textbf{(IV.15)}$$

$$\begin{aligned}\phi_5 = (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12}) \times \left[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2\right] + \\ (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03}) \times \left[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2\right]\end{aligned} \quad \textbf{(IV.16)}$$

$$\phi_6 = (\mu_{20} - \mu_{02})\left[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2\right] + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03}) \quad \textbf{(IV.17)}$$

$$\begin{aligned}\phi_7 = (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12}) \times \left[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2\right] - \\ (\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03}) \times \left[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2\right]\end{aligned} \quad \textbf{(IV.18)}$$

En plus de l'invariance aux changements d'échelle et de translation, les 7 moments de Hu définis plus haut sont aussi invariants à la rotation et la réflexion sauf le 7ème moment qui change de signe après réflexion.

Les 7 moments de Hu ainsi définis sont calculés pour les 3 canaux de couleurs que ce soit pour les canaux RVB ou HSV ; ce qui donne un vecteur caractéristique formé de 21 composantes. Ils peuvent être aussi calculés pour l'image en niveau de gris. Ceci donne comme résultat un vecteur descripteur formé seulement de 7 composantes.

IV.4.2.2 Moments de Zernike

Les Moments de Zernike sont le mappage d'une image sur un ensemble de polynômes complexes de Zernike. Du fait que ces polynômes sont orthogonaux les uns aux autres, les moments de Zernike peuvent représenter les propriétés d'une image sans aucune redondance ou chevauchement d'informations entre les moments [***143***]. Grâce à ces caractéristiques, les

moments de Zernike ont été utilisés comme caractéristiques dans de nombreuses applications [***140***].

Le calcul des moments de Zernike à partir d'une image d'entrée peut se faire à travers trois étapes: calcul des polynômes radiaux, calcul des polynômes de Zernike, et le calcul des moments de Zernike en projetant l'image sur les polynômes de Zernike.

La méthode d'obtention des moments de Zernike à partir d'une image d'entrée commence par le calcul des polynômes radiaux. Le polynôme radial à valeurs réelles est défini comme suit:

$$R_{p,q}(r) = \sum_{s=0}^{(p-|q|)/2} \frac{(-1)^s (p-s)!\, r^{p-2s}}{s!\left(\frac{p+|q|}{2}-s\right)!\left(\frac{p-|q|}{2}-s\right)!} \qquad \textbf{(IV.19)}$$

Avec $R_{p,q}(r) = R_{p,-q}(r)$.

p et q sont généralement appelés respectivement l'ordre et la répétition. L'ordre p est un entier non négatif, et la répétition q est un nombre entier satisfaisant $p-|q| = (pair)$ et $|q| \le p$.

Les polynômes radiaux vérifient les propriétés orthogonales pour la même répétition q,

$$\int_0^{2\pi}\int_0^1 R_{p,q}(r,\theta) R_{p',q}(r,\theta) r\, dr\, d\theta = \frac{\delta_{pp'}}{2(n+1)} \qquad \textbf{(IV.20)}$$

Où $\delta_{pp'}$ est le symbole de Kronecker défini par : $\delta_{pp'} = \begin{cases} 1 & if \quad p = p' \\ 0 & if \quad p \neq p' \end{cases}$

En utilisant les polynômes radiaux, les polynômes complexes 2-D de Zernike qui sont définis dans un cercle unité, sont donnés par:

$$V_{pq}(x,y) = V_{pq}(r\sin\theta, r\cos\theta) = R_{p,q}(r) e^{jq\theta} \qquad \textbf{(IV.21)}$$

Où, $j = \sqrt{-1}$, $|r| \le 1$ est la longueur du vecteur de l'origine du pixel en (x, y), et θ est l'angle entre le vecteur r et l'axe des x.

Les polynômes de Zernike sont un ensemble complet de fonctions complexes orthogonales sur le disque unité:

$$\iint_{x^2+y^2\le 1} [V_{mn}(x,y)]^* V_{pq}(x,y)dxdy = \frac{\pi}{m+1}\delta_{mp}\delta_{nq} \qquad \textbf{(IV.22)}$$

Ou, en coordonnées polaires:

$$\int_0^{2\pi}\int_0^1 [V_{mn}(r,\theta)]^* V_{pq}(r,\theta)rdrd\theta = \frac{\pi}{m+1}\delta_{mp}\delta_{pq} \qquad \textbf{(IV.23)}$$

Où l'astérisque désigne le complexe conjugué.

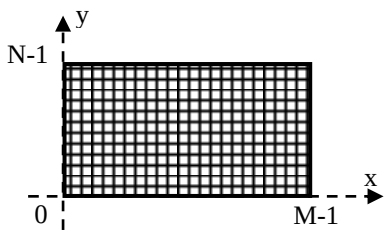
Les moments de Zernike complexes d'ordre p et de répétition q pour une fonction d'image f (x, y) sont finalement définis comme suit:

$$Z_{pq} = \frac{p+1}{\pi}\iint_{x^2+y^2\le 1} [V_{pq}(x,y)]^* f(x,y)dxdy \qquad \textbf{(IV.24)}$$

Ou bien, en coordonnées polaires:

$$Z_{pq} = \frac{p+1}{\pi}\int_0^{2\pi}\int_0^1 [V_{pq}(r,\theta)]^* f(r,\theta)r\,dr\,d\theta \qquad \textbf{(IV.25)}$$

Selon cette définition, la procédure pour calculer les moments de Zernike peut être considérée comme un produit scalaire entre la fonction de l'image et les polynômes de Zernike.



(a)

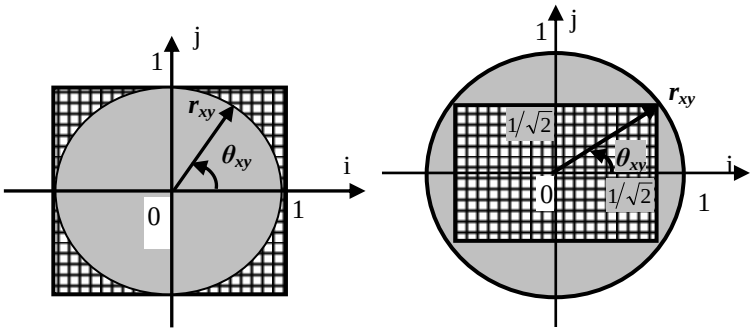


Figure IV-6 : Exemple d'image M x N (a), son mappage sur un cercle unité (b) et son mappage à l'intérieur d'un cercle unité (c).

Pour calculer les moments de Zernike d'une image numérique, les intégrales dans les équations (IV.24) et (IV.25) sont remplacées par des sommations en plus des coordonnées de l'image qui doivent être normalisées dans l'intervalle [0, 1] par une transformation de mappage. Les deux cas couramment utilisés pour ces transformations sont présentés dans la Figure IV-6: (b) l'image est sur un cercle de rayon unité, et (c) l'image est à l'intérieur d'un cercle unité. Sur la Figure IV-6 (b), les pixels, qui sont situés à l'extérieur du cercle, ne sont pas impliqués dans le calcul des moments de Zernike. En conséquence, les moments de Zernike, qui sont calculés par la transformation de mappage, ne décrivent pas les propriétés de l'extérieur du cercle unité dans l'image. Cela peut être considéré comme un défaut lors du calcul des moments de Zernike. La forme discrète des moments de Zernike d'une image de taille M x N est exprimée comme suit:

$$\begin{aligned} Z_{pq} &= \frac{p+1}{\lambda}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[V_{pq}(x,y)\right]^{*} f(x,y) \\ &= \frac{p+1}{\lambda}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}R_{pq}(r_{xy})e^{-jq\theta_{xy}} f(x,y) \end{aligned} \qquad \textbf{(IV.26)}$$

Où $0 \le r_{xy} \le 1$ et λ est un facteur de normalisation.

Dans la mise en œuvre discrète des moments de Zernike, le facteur de normalisation λ doit être le nombre de pixels, qui sont situés dans le cercle unité de la transformation de mappage, qui correspond à l'aire d'un cercle unité π dans le domaine continu. La phase transformée θ_{xy} et la distance transformée r_{xy} au niveau du pixel de coordonnées (x, y) sont données par:

Pour la transformation de la Figure IV-6 (b):

$$\theta_{xy} = \tan^{-1}\left(\frac{(2y-(N-1))/(N-1)}{(2x-(M-1))/(M-1)}\right) \quad \textbf{(IV.27)}$$

$$r_{xy} = \sqrt{\left(\frac{2x-(M-1)}{M-1}\right)^2 + \left(\frac{2y-(N-1)}{N-1}\right)^2} \quad \textbf{(IV.28)}$$

Et pour la transformation de la Figure IV-6 (c):

$$\theta_{xy} = \tan^{-1}\left(\frac{2y-(N-1)}{2x-(M-1)}\right) \quad \textbf{(IV.29)}$$

$$r_{xy} = \sqrt{\frac{(2x-(M-1))^2 + (2y-(N-1))^2}{(M-1)^2 + (N-1)^2}} \quad \textbf{(IV.30)}$$

L'Algorithme IV.4 montre le pseudo-code pour calculer les moments de Zernike d'ordre p et de répétition q par l'Equation (IV.26) et en utilisant la méthode directe pour les polynômes radiaux.

Algorithme IV.4 : Calcul des moments de Zernike par la méthode directe.

Algorithme : Calcul des moments de Zernike par la méthode directe.

Entrées : L'image $I = f(x,y)$ de taille $M \times N$, l'ordre p et la répétition q.

Sorties : Moments de Zernike complexe.

Début

Fonction RadialPolinomial (r, p, q)

$radial \leftarrow 0$;

Pour s=0 **à** $E((p-q)/2)$ **Faire**

$$c = \frac{(-1)^s (p-s)!}{s!\left(\frac{p+|q|}{2}-s\right)!\left(\frac{p-|q|}{2}-s\right)!} ;$$

$radial \leftarrow radial + c \times r^{p-2s}$;

Fin Pour;

Retourner radial ;

Fonction ZernikeMoments (p, q)

$Zr \leftarrow 0$; //Partie réelle.

$Zi \leftarrow 0$; //Partie imaginaire.

$Count \leftarrow 0$; //Compteur de pixel (surface)

Pour x=0 **à** (M-1) **Faire**

pour y=0 **à** (N-1) **Faire**

$$r = \sqrt{\left(\frac{2x-(M-1)}{M-1}\right)^2 + \left(\frac{2y-(N-1)}{N-1}\right)^2} \ ;$$

$$\theta = \tan^{-1}\left(\frac{(2y-(N-1))/(N-1)}{(2x-(M-1))/(M-1)}\right);$$

Si $(r \le 1)$ **Alors**

$Radial \leftarrow RadialPolynomial(r, p, q)$;

$Zr \leftarrow Zr + f(x,y) \times Radial \times \cos(q \times \theta)$;

$Zi \leftarrow Zi + f(x,y) \times Radial \times \sin(q \times \theta)$;

$Count \leftarrow Count + 1$;

Fin Si ;

Fin Pour;

Fin Pour;

retourner $\frac{(p+1)(Zr + i \times Zi)}{Count}$; *//Moments complexe de Zernike.*

Fin.

La plupart du temps de calcul des moments de Zernike est dû au calcul des polynômes radiaux. Par conséquent, les chercheurs ont proposé des méthodes plus rapides qui réduisent les termes factoriels en utilisant les relations de récurrence sur les polynômes radiaux.

Prata et al. [***144***] ont proposé la relation de récurrence qui utilise les polynômes radiaux de l'ordre inférieur à p comme suit:

$$R_{pq}(r) = \frac{2rp}{p+q} R_{(p-1)(q-1)}(r) - \frac{p-q}{p+q} R_{(p-2)q}(r) \qquad \textbf{(IV.31)}$$

Il est évident qu'à partir de l'équation précédente nous ne pouvons pas calculer tous les cas de p et q lors du calcul des polynômes radiaux. Il n'est pas possible d'utiliser l'équation de Prata dans le cas où q = 0 et p = q. Ces cas peuvent être obtenus par d'autres méthodes. Le calcul par la méthode directe peut être utilisée dans les cas où q = 0, tandis que l'équation $R_{pp}(r) = r^p$ est utilisée pour p = q. L'utilisation de la méthode directe pour calculer

les polynômes radiaux dans le cas où q = 0 augmentera considérablement le temps de calcul surtout si p est grande.

Kintner [***145***] a proposé une autre relation de récurrence qui utilise les polynômes à petit ordre p variable et une répétition q fixe pour calculer les polynômes radiaux comme illustré ci-dessous:

$$R_{pq}(r) = \frac{(K_2 r^2 + K_3) R_{(p-2)q}(r) + K_4 R_{(p-4)q}(r)}{K_1} \qquad \textbf{(IV.32)}$$

Où les coefficients K1, K2, K3 et K4 sont donnés par :

$$K_1 = \frac{(p+q)(p-q)(p-2)}{2}$$

$$K_2 = 2p(p-1)(p-2)$$

$$K_3 = -q^2(p-1) - p(p-1)(p-2)$$

$$K_4 = \frac{-p(p+q-2)(p-q-2)}{2}$$

A partir l'équation (IV.32), la méthode de Kintner ne peut être appliquée dans le cas où p = q et p - q = 2. Pour ces deux cas, il est préférable d'utiliser la méthode directe même si cela prend beaucoup de temps de calcul. Deux relations sont utilisées pour éviter l'implication de la méthode directe. Pour p = q on utilise l'équation $R_{pp}(r) = r^p$, et pour p - q = 2 on utilise:

$$R_{pq}(r) = pR_{pp}(r) - (p-1)R_{qq}(r) \qquad \textbf{(IV.33)}$$

Cette version améliorée de la méthode de Kintner est désignée comme la méthode de Kintner modifiée.

Récemment, Chong [***146***] a présenté la méthode q-récursif, qui utilise une relation de polynômes radiaux d'ordre p fixe et de répétition q variable. La relation du polynôme radial est définie comme suit :

$$R_{pq}(r) = H_1 R_{p(q+4)}(r) + \left(H_2 + \frac{H_3}{r^2}\right) R_{p(q+2)}(r) \qquad \textbf{(IV.34)}$$

Où :

$$H_1 = \frac{(q+4)(q+3)}{2} - (q+4)H_2 + \frac{H_3(p+q+6)(p-q-4)}{8}$$

$$H_2 = \frac{H_3(p+q+4)(p-q-2)}{4(q+3)} + (q+2)$$

$$H_3 = -\frac{4(q+2)(q+1)}{(p+q+2)(p-q)}$$

Comme l'ordre p est fixé dans l'équation (IV.34), l'ordre individuel des moments de Zernike peut être calculé de façon indépendante, sans référence au plus grand ou au plus petit ordre des moments.

Toutes ces précédentes méthodes de calcul se concentrent uniquement sur le calcul des polynômes radiaux de Zernike et présentent certaines limitations si un seul moment de Zernike est nécessaire, car ils utilisent les relations de récurrence. D'après les expériences dans [***146***], l'utilisation combinée à la fois de la méthode q-récursive et la méthode modifiée de Kintner prend moins de temps pour calculer un ensemble complet de moments de Zernike suivis par la méthode de Kintner. La méthode de Chong est beaucoup plus rapide que les autres méthodes en particulier pour le calcul des moments de Zernike avec un ordre fixe. Par conséquent, la méthode de Chong est plus efficace dans les cas où seul un ensemble d'ordres sélectionnés de moments de Zernike sont utilisés comme vecteur de caractéristiques.

La méthode de Kintner et la méthode q-récursif modifiées sont combinées et utilisées pour le calcul des polynômes de Zernike radiaux. Les moments de Zernike sont ainsi obtenus en utilisant la méthode hybride définie par :

$$\begin{cases} \begin{cases} R_{p,q}(r) = \sum_{s=0}^{(p-|q|)/2} \frac{(-1)^s (p-s)!\, r^{p-2s}}{s!\left(\frac{p+|q|}{2}-s\right)!\left(\frac{p-|q|}{2}-s\right)!} \\ if\ \ q=0,\ p-|q| = (even),\ \ |q| \le p,\ \ p \ge 0 \end{cases} \\ \begin{cases} R_{p,q}(r) = r^p \\ if \quad p = q \end{cases} \\ \begin{cases} R_{p,q}(r) = \frac{2rp\,R_{p-1,q-1}(r) - (p-q)R_{p-2,q}(r)}{p+q} \\ if\ \ p \ne q,\ q \ne 0 \end{cases} \end{cases} \qquad \textbf{(IV.35)}$$

L'utilisation de la méthode directe pour calculer les polynômes radiaux dans le cas de q = 0 augmentera considérablement le temps de calcul surtout quand p est très grand.

Les moments de Zernike ainsi définis sont calculés pour les 3 canaux de couleurs, que ce soit les canaux RVB ou HSV. Ils peuvent être aussi calculés pour l'image en niveau de gris.

IV.4.2.3 Moments de Legendre

Les moments de Legendre, ont été introduits par Teague [***142***]. Les moments de Legendre sont des moments orthogonaux. Ils ont été utilisés dans plusieurs modèles de reconnaissance de formes [***147***].

Le (p + q)-ième ordre des moments de Legendre d'une image avec la fonction d'intensité f (x, y) est défini sur le carré [-1,1] x [-1,1] et donné par l'équation suivante:

$$L_{pq} = \lambda_{pq} \int_{-1}^{+1}\int_{-1}^{+1} P_p(x)\, P_q(y)\, f(x,y)\, dxdy \qquad \textbf{(IV.36)}$$

Où $\lambda_{pq} = \frac{(2p+1)(2q+1)}{4}$, p,q=0,1,2,..,∞.

Et $P_p(x)$ désigne le P[th] ordre du polynôme de Legendre défini par:

$$P_p(x) = \sum_{k=0}^{p} a_{pk} x^k = \frac{1}{2^p\, p!} \frac{d^p}{dx^p}\left(x^2-1\right)^p \qquad \textbf{(IV.37)}$$

Ou encore,

$$P_p(x)=\sum_{k=0}^{p}\left\{\frac{(-1)^{\frac{p-k}{2}}x^k(p+k)!}{2^p k!\left(\frac{p-k}{2}\right)!\left(\frac{p+k}{2}\right)!}\right\}_{p-k=even} \quad \textbf{(IV.38)}$$

Les polynômes de Legendre ont comme fonction génératrice:

$$\frac{1}{\sqrt{1-2rx+r^2}}=\sum_{p=0}^{\infty}r^p P_p(x) \quad , \quad r\prec 1 \quad \textbf{(IV.39)}$$

En dérivant les deux parties de génération de fonction ci-dessus, la formule récurrente des polynômes de Legendre peut être acquise par:

$$\frac{d}{dr}\left(\frac{1}{\sqrt{1-2rx+r^2}}\right)=\frac{d}{dr}\left(\sum_{p=0}^{\infty}r^p P_p(x)\right) \quad \textbf{(IV.40)}$$

$$\frac{1}{\sqrt{1-2rx+r^2}}\times\frac{x-r}{1-2rx+r^2}=\sum_{p=0}^{\infty}pr^{p-1}P_p(x) \quad \textbf{(IV.41)}$$

Ensuite, nous avons:

$$(x-r)\sum_{s=0}^{\infty}r^p P_p(x)=(1-2rx+r^2)\sum_{p=0}^{\infty}pr^{p-1}P_p(x) \quad \textbf{(IV.42)}$$

Et la formule récurrente des polynômes de Legendre devient:

$$\begin{cases}P_{p+1}(x)=\dfrac{2p+1}{p+1}\,x\,P_p(x)-\dfrac{p}{p+1}P_{p-1}(x)\\ P_1(x)=x \quad , \quad P_0(x)=1\end{cases} \quad \textbf{(IV.43)}$$

Les polynômes de Legendre sont une base orthogonale complète définie sur l'intervalle [-1, 1]:

$$\int_{-1}^{+1}P_p(x)P_q(x)dx=\frac{2}{2p+1}\delta_{pq} \quad \textbf{(IV.44)}$$

Où $\delta_{pq} = \begin{cases} 1 & if \quad p = q \\ 0 & if \quad p \neq q \end{cases}$ est le symbole de Kronecker.

La propriété d'orthogonalité des polynômes de Legendre n'implique pas de redondance ou de chevauchement d'informations entre les moments d'ordres différents. Cette propriété permet la contribution de chaque moment, d'une manière unique et indépendante, à l'information contenu dans une image [***142***].

Pour calculer les moments de Legendre à partir d'une image numérique, les intégrales dans l'équation (IV.36) sont remplacées par des sommations et les coordonnées de l'image doivent être normalisées dans l'intervalle [-1, 1]. Par conséquent, la forme exacte des moments de Legendre, pour une image discrète de M x N pixels avec la fonction d'intensité f(x, y) est:

$$L_{pq} = \lambda_{pq} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} P_p(x_i)\, P_q(y_j)\, f(x, y) \qquad \textbf{(IV.45)}$$

Où $\lambda_{pq} = \dfrac{(2p+1)(2q+1)}{M \times N}$, x_i et y_j désignent les coordonnées normalisées des pixels dans l'intervalle [-1, 1], soit:

$$x_i = \frac{2x - (M-1)}{M-1} \quad et \quad y_j = \frac{2y - (N-1)}{N-1} \qquad \textbf{(IV.46)}$$

La formule définie dans l'équation (IV.45) est obtenue en remplaçant les intégrales dans l'équation (IV.36) par des sommations et en normalisant les coordonnées de pixels de l'image dans la plage [-1, 1] en utilisant l'équation (IV.46).

L'Algorithme IV.5 représente le pseudo-code pour le calcul des moments de Legendre d'ordre (p + q) à partir de l'équation (IV.45) et en utilisant la méthode directe de calcul de polynômes de Legendre. Dans ce travail, la formule récurrente est adoptée pour le calcul de polynômes de Legendre en vue d'augmenter la vitesse de calcul.

Algorithme IV.5 : Calcul des moments de Legendre par la méthode directe.

Algorithme : Calcul des moments de Legendre par la méthode directe.

Entrées: L'image $I = f(x, y)$ de taille $M \times N$ et l'ordre p et q;

Sorties: Moments de Legendre ;

Début

Fonction LegendrePolynomial (x, p)

$px \leftarrow 0$;

Pour k=0 **à** p **Faire**

Si $(Mod(p-k,2)=0)$ **Alors**

$$c = \frac{(-1)^{\frac{p-k}{2}} x^k (p+k)!}{2^p k! \left(\frac{p-k}{2}\right)! \left(\frac{p+k}{2}\right)!} ;$$

$px \leftarrow px + c$;

Fin Si

Fin Pour

Retourner px ;

Fonction LegendreMoments (p, q)

$L \leftarrow 0$; //Initialiser le moment de Legendre.

Pour x=0 **à** (M-1) **Faire**

Pour y=0 **à** (N-1) **Faire**

$$x_i = \frac{2x-(M-1)}{M-1} ;$$

$$y_i = \frac{2y-(N-1)}{N-1} ;$$

$px \leftarrow LegendrePolynomial(x_i, p)$;

$py \leftarrow LegendrePolynomial(y_i, q)$;

$L \leftarrow L + f(x,y) \times px \times py$;

Fin Pour

Fin Pour

Retourner $\frac{L(2p+1)(2q+1)}{M \times N}$; *//Moments de Legendre.*

Fin.

Les moments de Legendre définis précédemment sont calculés pour les 3 canaux de couleurs que ce soit les canaux RVB ou HSV. Ils peuvent être aussi calculés pour l'image en niveau de gris pour réduire la dimension du vecteur caractéristique.

IV.4.3 Texture

Plusieurs images présentent des motifs texturés. Dans ce cadre, il est essentiel de bien définir la texture et de s'intéresser plus particulièrement aux attributs qui la caractérisent.

IV.4.3.1 Notion et définition de la texture

Le petit robert [***148***] définit la texture comme étant un arrangement, une disposition des éléments d'une matière ou un agencement des éléments et parties d'une œuvre (d'un tout). La texture est décrite également par des termes linguistiques tels que la rugosité, le contraste, la finesse, la régularité [***149***]. Une texture présente, à une échelle donnée, le même aspect quelle que soit la zone observée. La Figure IV-7 montre quelques exemples de textures en couleur.

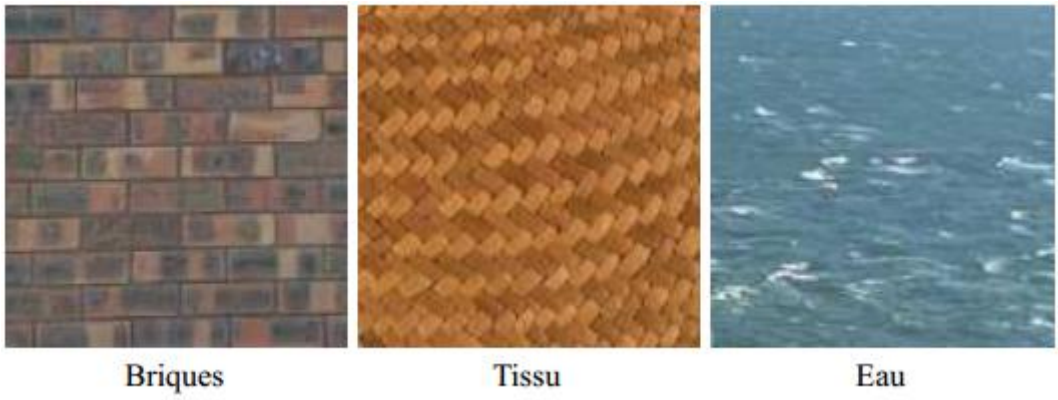


Figure IV-7 : Exemples de textures en couleur de type Briques, Tissu, et Eau

On rencontre deux types de texture [***150***] :

- Le premier type correspond à une vision microscopique : par exemple, la texture d'une brique est formée par l'arrangement des pixels représentant cette dernière. Les pixels sont donc les éléments structurants.

- Le second type correspond à une vision macroscopique : c'est à la texture du mur de briques que l'on s'intéresse dans ce cas et les éléments structurants sont cette fois-ci les briques elles-mêmes.

Le type de texture considéré va dépendre de la taille du voisinage utilisé pour extraire les attributs de texture.

IV.4.3.2 Principaux attributs de texture couleur

Depuis une trentaine d'années, les chercheurs se sont beaucoup intéressés à la classification d'images texturées. Que ce soit dans le domaine du contrôle de qualité industriel, ou encore dans celui de l'analyse de scènes, la texture est présente partout et de nombreuses méthodes ont été mises en place afin de satisfaire les besoins actuels en termes de classification et de segmentation d'images.

Après une brève description de quelques uns des attributs de texture les plus connus à ce jour, ceux basés sur les matrices de cooccurrences chromatiques ainsi que les indices d'Haralick extraits de ces matrices seront détaillés par la suite.

IV.4.3.2.1) Champs de Markov

L'utilisation des champs de Markov en traitement d'images comme en vision par ordinateur s'est développée depuis quelques années. Les principales applications de ce type de modélisation sont la détection de contours, la restauration ou le lissage d'images, et bien sûr la classification d'images couleurs texturées. Il existe donc de nombreuses extensions des champs de Markov, chacune d'entre elles étant spécifique à l'un des nombreux domaines d'application [***151***], [***152***].

Le plus souvent utilisée dans un cadre bayésien où l'on estime les paramètres du modèle et leur intervalle de confiance, cette approche consiste à modéliser non seulement les textures, mais aussi les déformations subies lors de l'acquisition. Plus précisément, les champs de Markov permettent une modélisation des pixels et des interactions spatiales entre ces derniers.

Une objection souvent faite à l'utilisation des champs de Markov est le coût en temps de calcul, ce temps étant fortement lié au type de modèle utilisé et à la complexité de la tâche à effectuer.

IV.4.3.2.2) Filtres de Gabor

Le but des filtres de Gabor est de sélectionner, dans le domaine de Fourier, l'ensemble des fréquences qui sont propres à chaque classe de textures [***153***].

Ils font partie des filtres les plus utilisés dans le domaine de la classification d'images couleur texturées. En particulier, ils sont utilisés pour l'analyse de séquences d'images car ils permettent d'intégrer et d'associer des informations spatiales et temporelles. Cette propriété permet de détecter des objets en mouvement. De plus, des études physiologiques ont montré qu'il est possible d'assimiler le fonctionnement de certains neurones du cortex visuel à ce type de filtres.

Pour la classification, on utilise une batterie de filtres de Gabor, qui est un ensemble de filtres, chacun étant sensible à une fréquence particulière. On applique alors chacun de ces filtres, un par un, à l'image couleur à classer, et on calcule à chaque étape l'énergie de l'image filtrée. Après avoir traité l'image avec tous les filtres de la batterie, on obtient un vecteur d'attributs de texture, dont les composantes sont les énergies calculées pour chaque fréquence [***154***].

L'inconvénient principal de cette méthode est le réglage des paramètres des filtres. De plus, dans certains cas, la taille des filtres nécessaires pour obtenir de bons résultats doit être assez grande ce qui implique un temps de calcul assez élevé.

IV.4.3.2.3) Ondelettes

Apparues au début des années 1980, les ondelettes s'imposent aujourd'hui comme des outils puissants en analyse mathématique et des applications telles que le traitement du signal et de l'image (la segmentation, le dé-bruitage, la compression d'images mais aussi l'analyse de textures couleur). Tout comme les filtres de Gabor, la transformée en ondelettes permet une représentation temps-fréquence. Il existe de nombreuses décompositions par ondelettes, chacune ayant ses spécificités : transformées en ondelettes orthogonales, bi-orthogonales, transformées discrètes, continues,… [***155***].

Pour extraire les attributs de texture, on considère la transformée en ondelettes de l'image. Cette transformée est en fait une matrice de coefficients, de taille similaire à l'image initiale, dont les attributs de texture seront extraits.

IV.4.3.2.4) Quaternions

Cet outil mathématique offre une nouvelle représentation de la couleur qui est souvent décrite par trois composantes C_1, C_2 *et* C_3 (R, G et B ou H, S et V par exemple). Les quaternions permettent de décrire la couleur non plus par un triplet, mais par un seul nombre dit hypercomplexe [***156***]. En effet, un quaternion q possède la forme suivante : $q = a + b \cdot i + c \cdot j + d \cdot k$, avec une partie réelle a et trois parties imaginaires b, c et d . Un triplet (C_1, C_2, C_3) d'une image couleur pourra donc être représenté par un quaternion purement imaginaire de la forme $C_1 \cdot i + C_2 \cdot j + C_3 \cdot k$. Ceci permet de représenter la couleur par une seule composante.

Pour obtenir les attributs de texture d'une image couleur, on calcule la matrice des quaternions conformément à la Figure IV-8 et on y applique la méthode d'analyse en composantes principales des quaternions (Quaternion principal component analysis en anglais) qui correspond à une analyse en composantes principales appliquée aux quaternions [***157***].

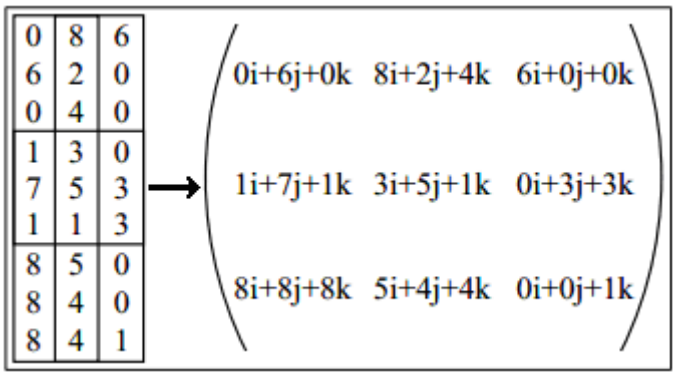


Figure IV-8 : Image couleur où la couleur de chaque pixel est représentée par une cellule (C_1,C_2,C_3) et sa matrice de quaternions.

IV.4.3.2.5) Matrices de cooccurrences

Aucun des attributs présentés précédemment ne permet d'obtenir des résultats satisfaisants en termes de classification d'images texturées quelque soient les textures présentes dans les images considérées. De nombreux chercheurs ont utilisé les matrices de cooccurrences pour l'analyse de texture couleur. Certains ont même utilisé les attributs d'Haralick et d'autres ont testé leur méthode d'analyse dans différents espaces couleur.

a) Matrices de cooccurrences à niveaux de gris

La matrice de cooccurrence d'une image est un outil statistique intéressant introduit par Haralick en 1973 [***158***], car il mesure la distribution des niveaux de gris dans l'image tout en prenant en compte les interactions spatiales entre les pixels.

Considérons $M_{[I]}$, la matrice de cooccurrences qui mesure l'interaction spatiale entre les pixels de l'image I. Le contenu de la cellule $M_{[I]}(i, j)$ de cette matrice indique le nombre de fois qu'un pixel p de l'image I, dont le niveau de gris est égal à j possède, dans son voisinage, un pixel voisin p' dont le niveau de gris est égal à i.

La notion de voisinage ou de proximité des pixels peut changer selon la direction considérée ainsi que la distance prise en compte d, la Figure IV-9 montre un exemple de proximité entre le pixel considéré et ses pixels voisins. L'orientation et la direction peuvent être définies en spécifiant le nombre de pixels considérés pour le déplacement, que ce soit dans la direction des x ou dans la direction des y. Ces pixels sont définis par le couple (k, l).

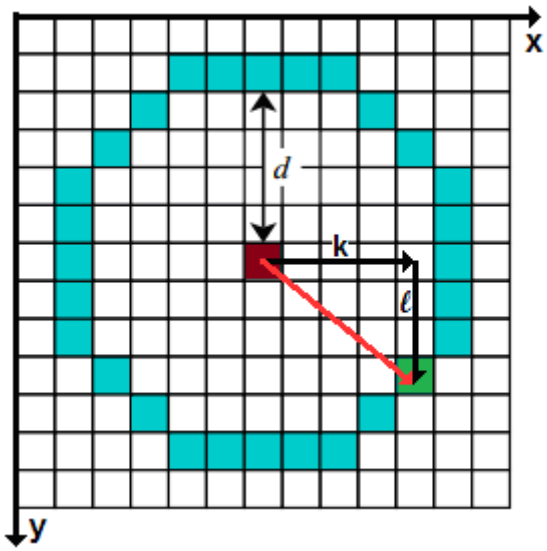


Figure IV-9 : Exemple de proximité entre le pixel considéré et ses pixels voisins.

Ainsi pour une image I de taille $N \times N$ et pour $(k, l) \in [1, \cdots, N]^2$ et $(a, b) \in [1, \cdots, G]^2$, la matrice de cooccurrence M_{kl} de l'image I est définie par :

$$M_{k,l}(a,b) = \frac{1}{(N-k)(N-l)} \sum_{i=1}^{N-k} \sum_{j=1}^{N-l} \delta(I(i, j) - a\ ,\ I(i+k, j+l) - b) \qquad \textbf{(IV.47)}$$

Dans cette équation, G indique le nombre de niveaux de gris dans l'image et δ représente l'impulsion unitaire de Dirac définie par :

$$\delta(x, y) = \begin{cases} 1 & si \quad x = 0 \quad et \quad y = 0 \\ 0 & \sin on \end{cases} \qquad \textbf{(IV.48)}$$

C'est une matrice de taille $G \times G$. Elle contient toute les statistiques du second ordre de l'image. Généralement, les valeurs de k *et* l sont petites devant la taille de l'image.

Comme pour les couleurs, il est inutile de conserver tous les niveaux de gris dans le calcul des textures. L'image peut être réduite par une quantification pour la faire passer de $G = 256$ niveaux à $T = 8, 16 \, ou \, 24$ niveaux de gris. Pour réduire les niveaux de gris, il suffit de diviser chaque pixel de l'image par T. Ainsi, l'image peut être quantifiée sur un nombre de niveaux de gris inférieur à G.

Les matrices de cooccurrences par image peuvent être calculées, pour une distance d=1 et pour des directions 0°, 45°, 90° et 135° degrés. Normalement, pour la direction 0 degré, $k = dx = 1$ et $l = dy = 0$; pour la direction 45 degrés, $k = dx = 1$ et $l = dy = 1$; pour la direction 90 degrés, $k = dx = 0$ et $l = dy = 1$ et pour la direction 135 degrés, $k = dx = -1$ et $l = dy = 1$.

Sharma a montré que les résultats obtenus par l'analyse des matrices de cooccurrences sont bons en termes de discrimination de textures en niveaux de gris [***159***]. Skrzypniak propose d'étendre le concept de matrices de cooccurrences aux images couleurs en définissant les matrices de cooccurrences chromatiques [***160***].

b) Matrices de cooccurrences chromatiques

Pour un espace de couleur quelconque noté (C_1, C_2, C_3) qui peut être l'espace de couleur (R, G, B) ou (H, S, V), et pour deux composantes couleurs $C, C' \in \{C_1, C_2, C_3\}$. La matrice de cooccurrences chromatique $M^{C,C'}[I]$ qui mesure l'interaction spatiale entre les composantes C et C' des pixels de l'image I. Le contenu de la cellule $M^{C,C'}[I](i, j)$ de cette matrice indique le nombre de fois qu'un pixel P de l'image I, dont le niveau de composante couleur $C^{C'}(p)$ est égal à j possède, dans son voisinage 3x3, un pixel p' voisin dont le niveau de composante $C^{C}(p')$ est égal à i.

Ainsi pour une image couleur I de taille $N \times N \times 3$ dans l'espace de couleur (C_1, C_2, C_3) et pour $(k, l) \in [1, \cdots, N]^2$ et $(a, b) \in [1, \cdots, G]^2$, la matrice de cooccurrence $M_{k,l}^{C,C'}[I]$ des deux composantes couleurs $C, C' \in \{C_1, C_2, C_3\}$ de l'image I est définie par :

$$M_{k,l}^{C,C'}([I], a, b) = \frac{1}{(N-k)(N-l)} \sum_{i=1}^{N-k} \sum_{j=1}^{N-l} \delta(I(i, j, C) - a, I(i+k, j+l, C') - b) \quad \textbf{(IV.49)}$$

Dans cette équation, δ représente l'impulsion unitaire définie par :

$$\delta(x, y) = \begin{cases} 1 & si \quad x = 0 \quad et \quad y = 0 \\ 0 & \text{sinon} \end{cases} \quad \textbf{(IV.50)}$$

C'est une matrice de taille $G \times G$. Elle contient toute les statistiques du second ordre de l'image. Généralement, les valeurs de k et l sont petites devant la taille de l'image.

Il est aussi inutile de conserver tous les niveaux des composantes de couleur dans le calcul des textures chromatiques. L'image peut être réduite par une quantification pour la faire passer de $G = 256$ à $T = 8, 16 \, ou \, 24$ niveaux. La réduction des niveaux de couleurs est effectuée en divisant chaque pixel du plan de l'image par T. Ainsi, l'image peut être quantifiée sur un nombre de niveaux de couleur inférieur à G.

La Figure IV-10 illustre ce propos. Elle représente une image couleur dans l'espace de couleur (C_1, C_2, C_3) ainsi que la matrice de cooccurrence $M^{C_1, C_2}[I]$ associée aux composantes de couleurs C_1, C_2 au milieu et la matrice de cooccurrence $M^{C_2, C_1}[I]$ associée aux composantes de couleurs C_2, C_1. Cette dernière matrice est obtenue par symétrie par rapport à la diagonale de la première matrice $M^{C_1, C_2}[I]$.

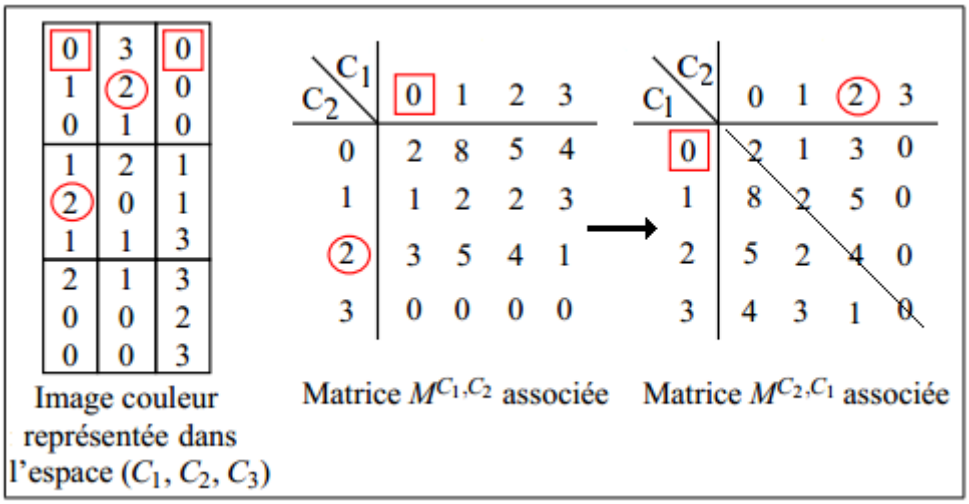


Figure IV-10 : Image couleur où la couleur de chaque pixel est représentée par une cellule (C_1,C_2,C_3) et ses deux matrices de cooccurrences chromatiques associées.

Ainsi par exemple, pour déterminer le contenu de la cellule située sur la troisième ligne et la première colonne de la matrice de cooccurrence $M^{C_1,C_2}[I]$ en milieu de la Figure IV-10, on examine le voisinage 3×3 des pixels de l'image couleur où $C_1 = 0$ (encadrés en rouge dans l'image couleur) et on compte le nombre de fois qu'on trouve, dans ce voisinage, des pixels dont la deuxième composante $C_2 = 2$ (cercles rouges dans l'image couleur). Pour le premier pixel où $C_1 = 0$ (en haut à gauche de l'image couleur) on a un pixel où $C_2 = 2$ sur la droite et un en dessous. Pour le deuxième pixel où $C_1 = 0$ (en haut à droite de l'image couleur) on a un pixel où $C_2 = 2$ sur la gauche, ce qui fait bien 3 en tout. La matrice de cooccurrence chromatique associée aux composantes inverses C_2 *et* C_1 est obtenue par symétrie par rapport à la diagonale de la matrice $M^{C_1,C_2}[I]$.

La possibilité de choisir entre différents types de voisinages peut être effectuée. En effet, l'étude des matrices de cooccurrences chromatiques peut se faire sur un voisinage 3×3 selon la 8-connexité ou selon la 4-connexité conformément au schéma de la Figure IV-11. Ce choix permet la comparaison de différentes approches et l'analyse de l'impact de ces approches sur la qualité des résultats obtenus. Il est même possible d'utiliser la notion de distance et de direction évoquée précédemment dans le calcul des matrices de cooccurrence en niveaux de gris en spécifiant un déplacement en grandeur de pixels dans la direction des x et un autre déplacement dans la direction des y par le couple $(k = dx, l = dy)$.

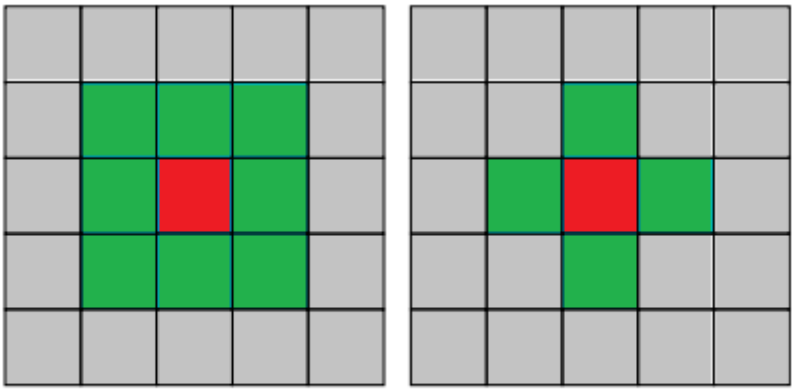


Figure IV-11 : Exemples de voisinage 3x3 selon la 8-connexité à gauche et la 4-connexité à droite.

Chaque image couleur I dans un espace de couleur (C_1, C_2, C_3), peut donc être caractérisée par 6 matrices de cooccurrences chromatiques :

- $M^{C_1, C_1}[I]$;
- $M^{C_2, C_2}[I]$;
- $M^{C_3, C_3}[I]$;
- $M^{C_1, C_2}[I]$;
- $M^{C_1, C_3}[I]$;
- $M^{C_2, C_3}[I]$.

Les matrices $M^{C_2, C_1}[I]$, $M^{C_3, C_1}[I]$ et $M^{C_3, C_2}[I]$ ne sont pas prises en compte car elles peuvent être déduites respectivement des matrices $M^{C_1, C_2}[I]$, $M^{C_1, C_3}[I]$ et $M^{C_2, C_3}[I]$ par symétrie diagonale. Ces matrices de cooccurrences chromatiques sont insensibles aux translations des objets dans les images et aux rotations dans un plan perpendiculaire à l'axe optique de la caméra. Comme elles mesurent les interactions locales entre les pixels, elles sont sensibles à des différences importantes de résolution spatiale entre les images. Pour atténuer cette sensibilité, il est nécessaire de normaliser ces matrices par le nombre total de cooccurrences dans la matrice considérée :

$$M_{k,l}^{C,C'}([I],a,b)=\frac{M_{k,l}^{C,C'}([I],a,b)}{\sum_{i=0}^{T-1}\sum_{j=0}^{T-1}M_{k,l}^{C,C'}([I],i,j)} \quad \textbf{(IV.51)}$$

Où T est le nombre de niveaux de quantification des composantes couleurs.

Les matrices de cooccurrences chromatiques permettent de caractériser les textures présentes dans les images [***161***]. Pourtant, elles ont un inconvénient majeur qui est leur coût important de stockage en mémoire. Afin de réduire le nombre d'informations, tout en conservant la pertinence de ces dernières, les indices d'Haralick [***158***] extraits de ces matrices sont utilisés (voir **Annexe B**).

Étant donné qu'il y aura 14 indices d'Haralick pour chaque matrice de cooccurrences, on aura 84 attributs de texture (14×6). Le nombre total d'attributs étant très important, il convient alors d'effectuer une sélection d'attributs afin de réduire la taille de l'espace de représentation des attributs.

IV.4.4 Descripteurs GIST

Dans la vision par ordinateur, les descripteurs de GIST sont une représentation d'une image en basse dimension qui contient suffisamment d'informations pour identifier la scène dans une image. Les descripteurs globaux GIST permettent d'avoir une représentation très faible de la dimension d'une image. Ces descripteurs ont été introduits par Oliva et Torralba en 2001 [***162***], [***163***]. Ils permettent de représenter la structure dominante spatiale de la scène à partir d'un ensemble de dimensions perceptives. Ils ne nécessitent aucune segmentation. Les auteurs ont essayé de capturer le descripteur de GIST de l'image en analysant les fréquences spatiales et les orientations tel que présenté dans la Figure IV-12. Le descripteur global est construit en combinant les amplitudes obtenues à la sortie de K filtres de Gabor [***164***] à différentes échelles E et orientations O. Pour réduire la taille, chaque image est redimensionnée et subdivisée en N * N blocs (N compris entre 2 et 16), ce qui donne un vecteur de dimension N * N * K * E * O. Cette dimension peut être encore réduite par une analyse en composantes principales (ACP), qui donne également les poids appliqués aux différents filtres [***165***].

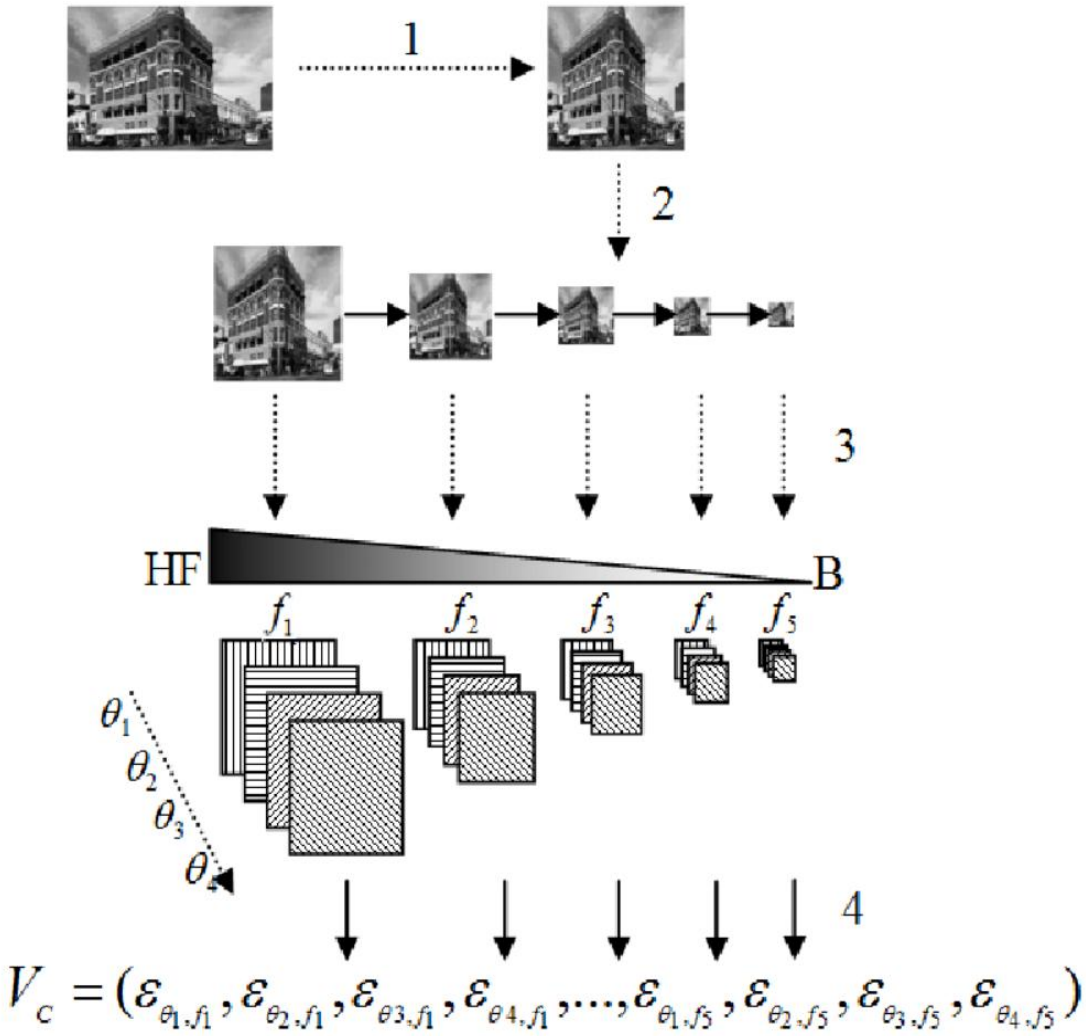


Figure IV-12 : Schéma de principe pour le calcul et l'extraction du descripteur de GIST.

Un filtre de Gabor est une fonction sinusoïdale modulée par une enveloppe gaussienne. La fonction sinusoïdale est caractérisée par sa fréquence et par son orientation. Ainsi un filtre de Gabor peut être vu comme un détecteur d'arêtes d'orientation particulières, puisqu'il réagit aux arêtes perpendiculaires à la direction de propagation du sinus [***166***]. Le filtrage par Gabor conserve les aspects temporels et fréquentiels du signal.

Un filtre de Gabor 2D est défini en continu de la façon suivante :

$$h(x, y) = g(x', y') \times \exp(2\pi i f x') \qquad \textbf{(IV.52)}$$

Avec :

$h(x, y)$ la réponse impulsionnelle du filtre.

$g(x', y')$ la fonction gaussienne 2D donnée par la formule suivante :

$$g(x', y') = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x'^2 + y'^2}{2\sigma^2}\right) \quad \textbf{(IV.53)}$$

avec $(x', y') = (x\cos\theta + y\sin\theta, -x\sin\theta + y\cos\theta)$ les coordonnées (x, y) tournées d'un angle θ.

Les paramètres f, θ et φ représentent la fréquence, l'orientation et la phase de la porteuse sinusoïdale et constituent les paramètres de l'espace du filtre de Gabor. Dans le domaine spatial, la fonction de Gabor est représentée par l'équation qui suit :

$$h(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \times \left(\cos(2\pi f x' + \varphi) + i\sin(2\pi f x' + \varphi)\right) \quad \textbf{(IV.54)}$$

Pour calculer les caractéristiques de Gabor dans le cas d'une gaussienne isotrope ($\gamma = 1$), un nombre de différents angles d'orientations $\theta \in [0, \pi]$ sont considérés et deux phases différentes $\varphi \in \left\{0, -\frac{\pi}{2}\right\}$ (0 pour le cas symétrique et $-\frac{\pi}{2}$ pour le cas anti symétrique).

Les paramètres les plus importants du filtre de Gabor sont la fréquence radiale et l'orientation, ils définissent la localisation du canal dans le plan fréquentiel.

Comme précisé ci-dessus, la Figure IV-12 présente un schéma résumant les différentes étapes pour le calcul et l'extraction de descripteurs de GIST. Après l'étape de prétraitement de l'image d'entrée, l'étape suivante consiste à modifier l'image en différentes échelles et orientations. Enfin, les vecteurs de caractéristiques de l'orientation et de la fréquence sont calculés pour chaque échelle. Ces vecteurs caractéristiques sont combinés pour former un descripteur global qui peut être réduit par une analyse en composantes principales (ACP).

IV.5 Annotation automatique d'images par classification

Nous commençons par définir le problème de classification. L'objectif de la classification est d'affecter à un objet, représenté par un vecteur d'un certain nombre de caractéristiques, une classe à partir de la base de données de référence. Pour classer les motifs et objets inconnus, un certain nombre d'échantillons d'entraînement, qui sont disponibles pour chaque classe, sont utilisés pour entrainer préalablement le classificateur qui permettra

de classer ces objets inconnus [***167***]. Cette tâche d'apprentissage ou d'entrainement est de calculer et définir un classificateur ou un modèle qui permet de faire de la correspondance entre les exemples d'entrée et leurs classes de sortie. L'étiquetage de ces exemples d'entrées doit être correct et l'entraînement doit être défini avec un certain niveau de précision. Cela peut être appelé l'entraînement ou l'étape de génération du modèle. Après avoir généré et entrainé le modèle, il est capable de classer une instance inconnue, comme étant une des classes définies et étiquetées dans l'ensemble d'entrainement. En d'autres termes, le classificateur calcule la similitude entre la classe inconnue et toutes les classes entrainées et assigne l'instance qui est inconnue sans étiquette à la classe ayant la plus grande similitude mesurée.

Par conséquent, l'annotation automatique d'images peut être abordée par le classificateur qui est généré et entrainé pour réduire l'écart entre le vecteur de bas niveau et les concepts de haut niveau. La fonction ainsi entrainée peut directement faire correspondre l'ensemble des caractéristiques de bas niveau aux classes théoriques de haut niveau.

On distingue deux grandes façons d'effectuer la classification :

- Modèle discriminatif.
- Modèle génératif.

Le modèle discriminatif permet de représenter les éléments que l'on souhaite classer par des points dans un espace à n-dimensions. L'apprentissage dans ce cas a pour objectif de tracer des frontières précises entre des ensembles de points représentant les catégories de classes. Le modèle génératif consiste à étudier d'abord la distribution des caractéristique et la probabilité à posteriori pour chacune des classes à classifier pour ensuite pouvoir inférer la probabilité à posteriori qui est utilisée dans la décision . Nous résumons ici les avantages et défauts de chacune de ces deux approches.

Pour les approches génératives:

- Elles peuvent prendre en compte des données manquantes ou partiellement annotées.
- De nouvelles classes sont faciles à ajouter, il suffit d'en apprendre le modèle.

- Les modèles génératifs peuvent simplement prendre en compte la composition (rajout de lunettes, d'un chapeau ou de moustaches à un visage) alors que les modèles Discriminatifs doivent recevoir tous les exemples possibles durant l'entrainement.

Pour les approches discriminatives :

- Le modèle génératif observe uniquement la façon dont les exemples d'une classe sont construits alors que les modèles discriminatifs se concentrent sur les aspects qui distinguent cette classe des autres, ce qui est potentiellement plus efficace pour prendre une décision.

- Les modèles discriminatifs sont le plus souvent très rapides pour classer un nouvel élément alors que les modèles génératifs ont souvent besoin d'un certain nombre d'itérations.

- Un modèle discriminatif étant spécifiquement entraîné pour prendre une décision entre plusieurs classes, il devrait être plus efficace pour un modèle créé pour déterminer seulement la présence conjointe d'éléments constituants d'une classe.

Il existe plusieurs types de classificateurs, chacun est jugé apte à classer un certain type particulier de vecteurs de caractéristiques en fonction de leurs paramètres. Les réseaux de neurones, les réseaux bayésiens, les machines à vecteurs de support (SVM) et les classificateurs à K-plus proches voisin sont les plus utilisés. Parmi ces classificateurs, on distingue ceux qui sont discriminatifs, à savoir les K-plus proches voisin, les réseaux de neurones et les SVM multiclasses, et ceux qui sont génératifs tels que les réseaux bayésiens.

Dans cette partie nous exposerons d'abord les différentes approches de classifications utilisées pour l'annotation automatique d'images. Ensuite, les algorithmes et le formalisme mathématique et algorithmique seront présentés.

IV.5.1 K-plus proches voisins

En intelligence artificielle, la méthode des k-plus proches voisins est une méthode d'apprentissage supervisé. En abrégé k-NN ou KNN, de l'anglais k-nearest Neighbors. C'est l'une des plus anciennes, plus simples et plus intuitives méthodes de classement [***168***], [***169***].

Elle peut se résumer par ce simple principe qui fait que des entrées semblables ou proches doivent avoir la même classe d'appartenance.

IV.5.1.1 Principe

Le principe intuitif de l'algorithme des K plus proches voisins repose sur les étapes suivantes [***168***]:

1. stocker les exemples tels quels dans une table;
2. pour prédire la classe d'une donnée, déterminer les exemples qui en sont le plus proche;
3. de ces exemples, déduire la classe de la donnée considérée.

Il faut noter l'utilisation d'une notion de proximité entre les exemples à l'étape 2. C'est la mesure de similarité entre un couple de données en utilisant une mesure de distance adéquate qu'on va éclaircir.

IV.5.1.2 Mesure de similarité entre deux données

La notion de proximité citée précédemment induit celle de distance. Cette notion formalise celle de similarité (dissemblance ou ressemblance) entre les données. Mathématiquement parlant, une distance d est une application définie par :

$$d: \quad ID \times ID \subset IR^2 \rightarrow IR^+$$
$$(x, y) \rightarrow d(x, y)$$

Qui doit respecter les trois propriétés suivantes $\forall (x, y, z) \in ID^3 \subset IR^3$:

- $d(x, y) = 0 \Leftrightarrow x = y \quad (identité\ ou\ séparabilité)$
- $d(x, y) = d(y, x) \quad (symétrie)$
- $d(x, z) \leq d(x, y) + d(y, z) \quad (inégalité\ triangulaire)$

La dernière propriété signifie que le chemin le plus court entre deux points ne peut être que le chemin direct entre les deux points. Ces trois propriétés doivent être respectées par une application de mesure de distance. Cependant, parfois, on n'exige que les deux premières

propriétés. L'inégalité triangulaire n'est pas toujours exigée surtout pour une application qui associe un réel positif à un couple de données telle que la similarité ou disimilarité.

Il nous faut être capable de mesurer la similarité entre deux données. Notons x_i et x_j deux données ayant p attributs. Pour des données dont les attributs sont tous quantitatifs, des mesures de distance souvent utilisées sont :

- Distance de Manhattan:

$$L_1(x_i, x_j) = \sum_{k=1}^{k=p} |x_{i,k} - x_{j,k}| \quad \textbf{(IV.55)}$$

- Distance Euclidienne :

$$L_2(x_i, x_j) = \left(\sum_{k=1}^{k=p} (x_{i,k} - x_{j,k})^2 \right)^{\frac{1}{2}} \quad \textbf{(IV.56)}$$

- Distance de Minkowski :

$$L_q(x_i, x_j) = \left(\sum_{k=1}^{k=p} |x_{i,k} - x_{j,k}|^q \right)^{\frac{1}{q}} \quad \textbf{(IV.57)}$$

- Distance de Tchebychev:

$$L_\infty(x_i, x_j) = \lim_{q \to \infty} \left(\sum_{k=1}^{k=p} |x_{i,k} - x_{j,k}|^q \right)^{\frac{1}{q}} = \max_{1 \le k \le p} |x_{i,k} - x_{j,k}| \quad \textbf{(IV.58)}$$

La Distance de Minkowski Lq représente la définition générale de la distance. En effet, la distance de Manhattan est obtenue dans le cas où $q = 1$, et pour $q = 2$ on trouve la distance Euclidienne.

On peut distinguer et discuter le cas d'une donnée à valeurs ou attributs numériques et le cas d'une autre donnée à attributs nominaux.

IV.5.1.2.1) Donnée à attributs numériques

Pour une donnée à attributs numériques, la similarité introduite plus haut s'applique immédiatement. Cependant, un problème se pose si l'ordre de grandeur des attributs à

combiner n'est pas le même pour tous ; naturellement, les attributs d'ordre de grandeur plus grand vont dominer les autres. Pour éviter ce problème, une mise à l'échelle est alors préférable. Il faut mettre à l'échelle les attributs en normalisant leurs valeurs. Si l'attribut x_i prend sa valeur dans l'intervalle $[\min(x_i), \max(x_i)]$, On utilise l'attribut normalisé suivant:

$$\hat{x}_i = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)} \in [0, 1] \quad \textbf{(IV.59)}$$

D'autre part, certains attributs peuvent être plus importants que d'autres. On peut donc pondérer les attributs en fonction de leur importance.

IV.5.1.2.2) Donnée à attributs nominaux

Du fait que les valeurs ne sont pas numériques, une mesure de similarité doit être définie. On prend généralement :

- une similarité nulle si les valeurs de l'attribut sont les mêmes dans les deux données;
- similarité = 1 sinon.

Ainsi, on n'a pas à normaliser la valeur des attributs nominaux. Cependant, cette définition peut être un peu trop simple et n'est pas adaptée aux attributs ordinaux ou nominaux. Ainsi, considérons un attribut « couleur » dont les valeurs possibles sont {rouge, jaune, rose, vert, bleu}. On peut se dire que la similarité entre bleu et rouge est plus grande qu'entre rouge et jaune. On peut alors définir une mesure de similarité qui en tienne compte.

IV.5.1.3 Algorithme des k plus proches voisins

Après avoir précisé la notion de proximité utilisée pour déterminer les plus proches voisins, on peut alors détailler le principe de l'algorithme. La détermination des k plus proches voisins est donnée avec plus de détails dans l'Algorithme IV.6.

Dans cet algorithme, la fonction de mesure de similarité est à définir avec une grande attention. Dans notre cas, puisque les données sont tous à attributs numériques homogènes ne nécessitant aucune normalisation ou mise à l'échelle, on a utilisé la distance euclidienne comme mesure de similarité.

Après avoir déterminé les k plus proches voisins, il reste à prédire la classe de la donnée x. cette classe ne peut être que la classe majoritaire des k plus proches voisins déterminés. Plusieurs stratégies sont envisageables. La plus simple est de choisir la classe majoritaire par un simple vote. L'autre stratégie consiste à déterminer la classe majoritaire parmi les k plus proches voisins en pondérant la contribution de chacun par une fonction qui peut être l'inverse de sa similarité à x, ou une autre fonction comme la fonction normale définie par :

$$f(a) = e^{-a^2} \qquad \textbf{(IV.60)}$$

Algorithme IV.6 : Prédiction de la classe d'une donnée par la méthode des k plus proches voisins.

Algorithme : Prédiction de la classe d'une donnée par la méthode des k plus proches voisins.

Entrées : un ensemble d'exemples $X \in D$ à N classes et une donnée $x \in D$ à p attributs; le nombre de proches voisins $1 \le k \le N$.

Sorties : la classe C_l de x tel que $l \in \{1,...,N\}$.

Début

pour chaque exemple $x_i \in X$ **faire**

calculer la similarité entre x_i *et* x : $S(x_i\ ,\ x) \leftarrow \left(\sum_{j=1}^{j=p} \left(x_{i,j} - x_j\right)^2 \right)^{\frac{1}{2}}$;

fin pour ;

pour $i \in \{1,...,k\}$ **faire**

$kppv[i] \leftarrow \arg\min_{j \in \{1,...,N\}} S(x_j\ ,\ x)$;

$poids[i] \leftarrow 1/S(x_{kppv[i]}, x)$ *ou bien* $poids[i] \leftarrow \exp\left(-\left(S(x_{kppv[i]}, x)\right)^2\right)$;

$S(x_{kppv[i]}\ ,\ x) \leftarrow +\infty$ *ou bien une très grande valeur* ;

fin pour ;

déterminer la classe C_l de x à partir de la classe majoritaire des k exemples dont le numéro l est stocké dans le tableau kppv par un maximum de votes ou par un maximum de pondération de similarité stocké dans le tableau poids.

Retourner la classe C_l ;

Fin.

IV.5.1.4 Choix du nombre de voisins k

Il n'y a pas de méthode particulière pour faire un choix optimum de k, il est choisit arbitrairement. Néanmoins, et d'une manière générale, on peut aussi prendre k = N dans le cas où l'influence de chaque exemple est pondérée par sa similarité avec la donnée à classer. Dans la littérature [***170***], on trouve la racine carrée du nombre d'individus pour le choix de k. Dans notre cas, on a choisit k=1.

IV.5.2 Réseaux de neurones

IV.5.2.1 Historique

Imiter et dupliquer l'intelligence humaine, constitue sans aucun doute le rêve ultime des chercheurs dans le domaine de l'intelligence artificielle. Les premiers travaux sur les Réseaux de Neurones Artificiels (RNA) ont été menés par Mc Culloch et Pitts en 1943 [***171***]. Ils montrent qu'il est possible de construire des systèmes vérifiant la définition de Turing pour les machines à calculer et donc capables de calculer des fonctions logiques.

En 1949, Donald Hebb [***172***] a établi la loi ou la théorie des assemblées de neurones. C'est une règle d'apprentissage des réseaux de neurones artificiels dans le contexte de l'étude d'assemblées de neurones. Cette règle aujourd'hui connue sous le nom de « règle de Hebb » suggère que lorsque deux neurones sont excités conjointement, un lien les unissant se crée ou se renforce. Elle décrit la manière dont les cellules apprennent à modifier les poids des connexions qui les relient. Elle est à la fois utilisée comme hypothèse en neurosciences et comme concept dans les réseaux neuronaux en mathématiques. En 1957, l'intégration de la loi de Hebb tout en tenant compte de l'erreur observée en sortie a permis le succès au perceptron de Franck Rosenblatt [***173***]. Ce modèle de perceptron constitue le premier système artificiel capable d'apprendre par expérience. Dans les années 50, les chercheurs ont commencé à développer des modèles non pas seulement software mais aussi hardware pour simuler les fonctions de bas niveaux du système nerveux [***174***].

Malgré tous ces efforts, le triomphe des RNA a connu une chute dans les années 70. Un coup grave fut porté à la communauté scientifique gravitant autour des réseaux de neurones. En effet, En 1969, Marvin Lee Minsky et Seymour Papert publièrent un ouvrage [***175***] mettant en exergue quelques limitations théoriques du Perceptron Rosenblatt, et plus généralement des classificateurs linéaires, notamment l'impossibilité de traiter des problèmes non linéaires ou de connexité. Mais au début des années 80, plusieurs chercheurs ont permis

de relancer les travaux de recherche dans ce domaine. En 1982, le physicien John Joseph Hopfield [***176***] donna un nouveau souffle aux réseaux de neurones en publiant un article introduisant un nouveau modèle de réseau de neurones qui est complètement récurrent. Ce modèle souffre de l'impossibilité de traiter les problèmes non-linéaires.

Une nouvelle génération de réseaux de neurones, capables de traiter avec succès des phénomènes non-linéaires et ne possédant pas les défauts mis en évidence par Marvin Minsky, a été proposée initialement pour la première fois par Werbos [***177***], [***178***] qui est le premier à étudier la rétro-propagation de l'erreur. Ce Perceptron multicouches introduit simultanément en détail par Rumelhart [***179***] et Yann [***180***] en 1986 reposent sur la rétro-propagation du gradient de l'erreur dans ses différentes couches.

Les réseaux de neurones ont par la suite connu un essor considérable. Actuellement l'utilisation des réseaux de neurones devient une réalité et les recherches continuent. Plusieurs articles ont été publiés sur les apports réciproques des réseaux de neurones [***174***], [***181***], [***182***], [***183***], [***184***].

Les réseaux neuronaux artificiels constituent alors une tentative de modélisation des réseaux de neurones biologiques d'où émergent les associations implicites du cerveau. Selon les chercheurs et les experts en intelligence artificielle, transposer leur structure dans des systèmes informatiques permettrait aux programmes d'acquérir par eux-mêmes des connaissances implicites. Une bonne compréhension du fonctionnement des neurones biologiques permet de mieux saisir celle de leurs analogues informatiques.

IV.5.2.2 Neurone biologique

La physiologie du cerveau montre que celui-ci est constitué de cellules (les neurones) interconnectées. Un neurone biologique est une cellule nerveuse constituant notamment l'élément de base du cerveau. Le neurone biologique représenté par la Figure IV-13 comprend :

- Le corps cellulaire, qui fait la somme des influx qui lui parviennent. Si cette somme dépasse un certain seuil, il envoie lui-même un flux à partir de l'axone.
- L'axone, qui permet de transmettre les signaux émis par le corps cellulaire aux autres neurones.

- Les dendrites qui sont les récepteurs principaux du neurone, captant les signaux qui lui parviennent.
- Les synapses qui permettent aux neurones de communiquer avec les autres via les axones et les dendrites.

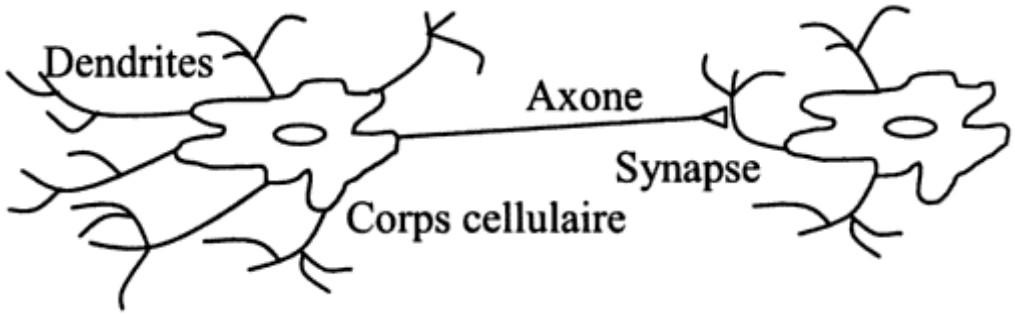


Figure IV-13 : Modèle du neurone biologique.

Les neurones reçoivent les signaux (impulsions électriques) par des extensions très ramifiées de leur corps cellulaire (les dendrites) et envoient l'information par de longs prolongements (les axones). Les impulsions électriques sont régénérées pendant le parcours le long de l'axone. Chaque neurone intègre en permanence jusqu'à un millier de signaux synaptiques. Ces signaux n'opèrent pas de manière linéaire : il y a un effet de seuil.

Comme il a été déjà précisé, les structures à la base du neurone biologique peuvent être modélisées sous la forme de neurones artificiels.

IV.5.2.3 Neurone artificiel

Un réseau de neurones artificiels est un modèle de calcul dont la conception et la structure sont très schématiquement inspirées du fonctionnement des neurones biologiques. Les principales structures biologiques des neurones ont toutes leurs équivalents artificiels. Le but d'une telle analogie est de reproduire le fonctionnement du neurone biologique de la meilleure façon possible, d'une manière logique, simple et facilement représentable sur ordinateur. Un neurone biologique est une minuscule structure qui traite les influx nerveux qui arrivent, chacun selon son importance relative, et qui émet un signal de sortie. Les neurones artificiels reproduisent le même procédé, recevant chaque signal d'entrée pondéré par un poids. Par analogie, ces poids sont aussi appelés poids synaptiques. Les entrées pondérées, habituellement sommées, sont ensuite comparées à un seuil d'activation θ et passées dans la fonction d'activation du neurone, qui produit la sortie désirée. Dans les

neurones biologiques, les entrées et les sorties sont des signaux électriques, représentés artificiellement par des valeurs numériques. Les neurones sont reliés ensemble en réseaux de plusieurs niveaux hiérarchiques appelés couches. Les neurones d'une couche passent leurs sorties aux neurones de la couche suivante et ainsi de suite. D'une manière générale, un neurone formel constitue un minuscule processeur, capable de gérer une fonction simple. C'est l'élément de base d'un réseau de neurones.

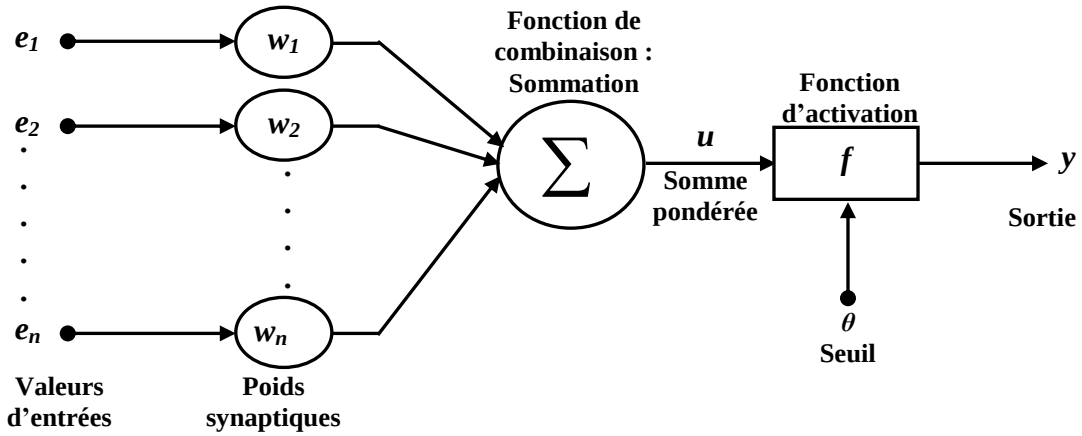


Figure IV-14 : Modèle du neurone artificiel.

La Figure IV-14 présente le modèle de fonctionnement d'un neurone formel élaboré par W. McCulloch et W. Pitts [***171***]. Il s'agit d'un composant calculatoire faisant la combinaison ou généralement la somme pondérée des signaux reçus en entrée (calculant la quantité u) à laquelle on applique une fonction de transfert et un seuil afin d'obtenir la réponse de la cellule (notée y). La sortie du neurone formel est donné par :

$$y = f\left(\sum_{i=1}^{n} w_i.e_i - \theta\right) \qquad \textbf{(IV.61)}$$

Tel que :

- e_i : élément du vecteur d'entrée,
- θ : le seuil du neurone formel (le seuil peut être ignoré),
- y : la sortie du neurone formel,
- w_i : les poids synaptiques de pondération du neurone formel,
- f : la fonction d'activation.

IV.5.2.4 Fonction de combinaison

Le neurone formel reçoit en amont un certain nombre de valeurs d'entrées via ses connexions ou poids synaptiques, et il produit à sa sortie une certaine valeur en utilisant une fonction de combinaison. Cette fonction peut donc être formalisée comme étant une fonction transformant un vecteur en un scalaire. Plusieurs fonctions de combinaison peuvent être utilisées. L'une des fonctions les plus utilisées est la combinaison linéaire des entrées, c'est-à-dire que la fonction de combinaison renvoie le produit scalaire entre le vecteur des entrées et le vecteur des poids synaptiques. L'autre fonction consiste à calculer la distance entre les entrées, c'est-à-dire que la fonction de combinaison renvoie la norme euclidienne du vecteur issu de la différence vectorielle entre les vecteurs d'entrées.

IV.5.2.5 Fonction d'activation

La fonction d'activation ou la fonction de seuillage, ou encore la fonction de transfert sert à introduire une non-linéarité dans le fonctionnement du neurone. Plusieurs fonctions peuvent être utilisées comme fonction de transfert pour l'activation d'un neurone. Les fonctions d'activations les plus utilisées et les plus célèbres sont les suivantes :

- Fonction échelon, ou fonction tout ou rien ;
- Fonction signe ;
- Fonction plus au moins à un seuil ;
- Fonction linéaire ou affine ;
- Fonction saturation ou fonction linéaire bornée ;
- Fonction sigmoïde ;
- Fonction tangente hyperbolique ;
- Fonction gaussienne.

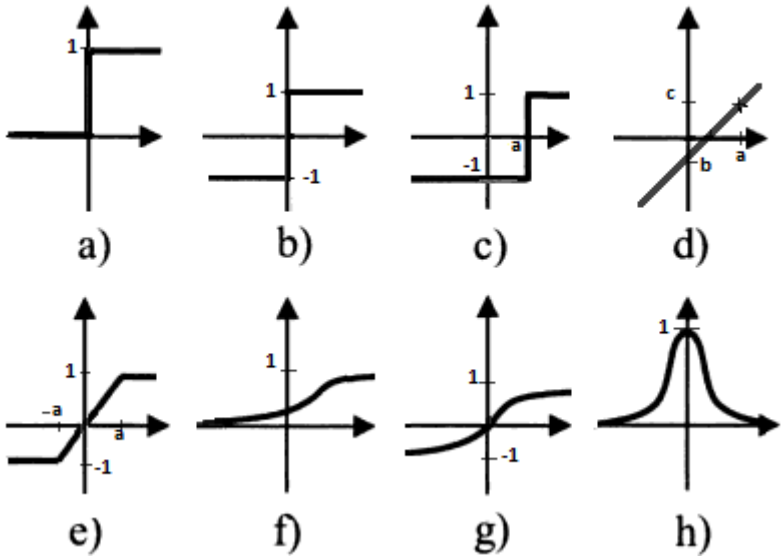
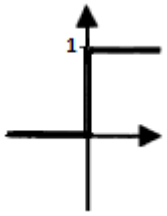
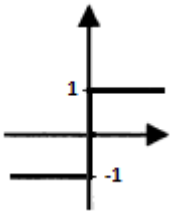
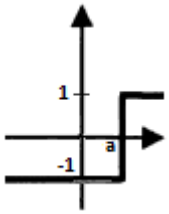
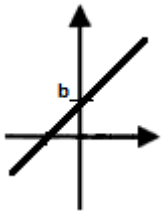
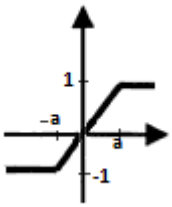
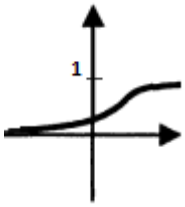
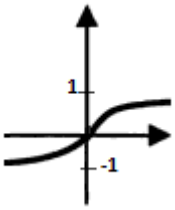
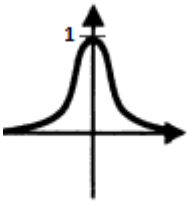


Figure IV-15 : Allure des courbes des fonctions de transfert des réseaux de neurones.

La Figure IV-15 précédente donne l'allure de leurs courbes en ordres tel qu'elles sont citées précédemment tandis que le Tableau IV-1 suivant donne leurs définitions.

Tableau IV-1 : Définitions et courbes des fonctions de transferts des réseaux de neurones.

Fonction	Courbe	Définition
Fonction Echelon		$f(x)=\begin{cases}0 & si \quad x \prec 0\\ 1 & si \quad x \geq 0\end{cases}$
Fonction signe		$f(x)=\begin{cases}-1 & si \quad x \prec 0\\ +1 & si \quad x \geq 0\end{cases}$
Fonction signe plus au moins à un seuil a		$f(x)=\begin{cases}-1 & si \quad x \prec a\\ +1 & si \quad x \geq a\end{cases}$

Fonction affine ou (fonction linéaire si b=0, identité si a=1 et b=0)		$f(x) = ax + b$
Fonction saturation ou fonction linéaire bornée		$f(x) = \begin{cases} bx & si \quad \lvert x \rvert \prec a \\ +1 & si \quad \lvert x \rvert \geq a \end{cases}$
Fonction Sigmoïde		$f(x) = \frac{1}{1+e^{-x}}$
Fonction tangente hyperbolique		$f(x) = 2/(1+\exp(-2x))-1$
Fonction Gaussienne		$f(x) = \exp\left(\frac{-x^2}{2}\right)$

Le choix de la fonction d'activation dépend de l'application. Toutes les fonctions d'activation utilisées doivent être dérivables, car l'architecture et la structure du réseau de neurones l'impose pour que l'apprentissage soit possible.

IV.5.2.6 Structure d'un réseau de neurones

L'architecture et la topologie d'un réseau de neurone artificiel est définie par l'organisation spatio-temporel des neurones formels qui le constituent. Une telle organisation constitue sa structure. Selon cette organisation, on trouve plusieurs types de réseaux de neurones tels que les réseaux monocouches et les réseaux multicouches.

IV.5.2.6.1) Réseau monocouche

C'est un perceptron ou réseau, formé généralement au plus, d'une couche de neurones formels. Ainsi qu'il a été souligné par Misnsky et Papert, Ce genre de perceptron permet de ne séparer que des exemples linéairement séparables. Par exemple, il ne peut pas réaliser le OU exclusif (XOR). Les réseaux multicouches permettent de palier à cette limitation.

IV.5.2.6.2) Réseau multicouche

Un réseau multicouche se compose d'une couche d'entrée comprenant un ensemble de nœuds d'entrée, éventuellement une ou plusieurs couches cachées de nœuds, et une couche de nœuds de sortie. La Figure IV-16 montre un exemple de réseau de neurones à trois couches, comportant une couche d'entrée formé par M nœuds, une couche cachée formée par L nœuds, et la couche de sortie formés par N nœuds.

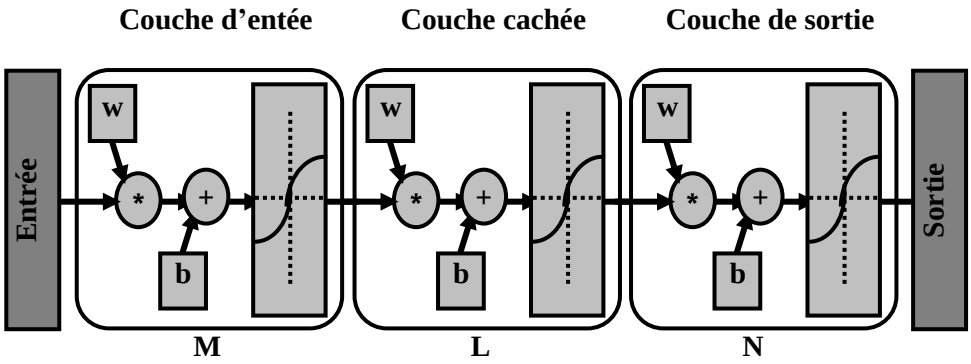


Figure IV-16 : Exemple de réseau de neurones à trois couches.

Ce réseau neuronal est entraîné à classer les entrées en fonction des catégories de classes cibles. Les données d'entrée d'apprentissage sont chargées à partir de la base de données de référence, tandis que les données cibles doivent consister en des vecteurs de l'ensemble des valeurs nulles à l'exception de l'un des éléments, où l'indice est la classe à représenter. La fonction de transfert utilisée dans ces trois couches du réseau de neurones est la tangente hyperbolique définie par:

$$f(x) = 2/(1+\exp(-2x)) - 1 \qquad \textbf{(IV.62)}$$

Selon les auteurs dans [***185***], le nombre de neurones dans la couche cachée est approximativement égale à:

$$L = E\left(1 + \sqrt{M\left(N+2\right)}\right) \quad \textbf{(IV.63)}$$

Ou E (x) désigne la partie entière de x.

Le nombre de nœuds d'entrée est fixé par le nombre de composantes du vecteur descripteur et celui de sortie est fixé par le nombre de classes utilisées lors de l'apprentissage du réseau de neurones.

IV.5.2.6.3) Apprentissage et entrainement du réseau de neurones

Une fonction des réseaux de neurones formels, à l'instar du modèle vivant, est d'opérer rapidement des classifications et d'apprendre à les améliorer. À l'opposé des méthodes traditionnelles de résolution informatique, on ne doit pas construire un programme pas à pas en fonction de la compréhension de celui-ci. Les paramètres importants de ce modèle sont les coefficients synaptiques, le seuil de chaque neurone et la façon de les ajuster. Ce sont eux qui déterminent l'évolution du réseau en fonction de ses informations d'entrée. Il faut choisir un mécanisme permettant de les calculer et de les faire converger si possible vers une valeur assurant une classification aussi proche que possible de l'optimale. C'est ce qu'on nomme la phase d'apprentissage du réseau. Dans un modèle de réseaux de neurones formels, apprendre revient donc à déterminer les coefficients synaptiques les moins mal adaptés à classifier les exemples présentés.

Ainsi, les réseaux de neurones artificiels apprennent par l'expérience, généralisent de nouvelles expériences à partir des expériences antérieures, et peuvent alors prendre des décisions [***119***], [***184***].

Après l'initialisation aléatoire des biais et des poids des connexions, le principe d'entraînement du réseau neuronal est basé sur une boucle d'étapes. Il commence par la propagation des entrées provenant de la couche d'entrée vers la couche de sortie. L'erreur calculée à la sortie du réseau est propagée en arrière pour chaque couche du réseau de neurones afin de mettre à jour les biais et les poids des connexions. Lorsque les biais et les poids des connexions sont modifiés, la propagation des entrées est répété jusqu'à avoir le minimum d'erreur entre les sorties et les cibles. C'est le critère d'arrêt de l'entraînement des réseaux de neurones.

Les entrées Y_i sont présentés à la couche d'entrée et se propagent vers la couche cachée à l'aide de la formule suivante:

$$Y_j = f\left(\sum_{i=1}^{m} Y_i w_i + b_i\right) \quad \textbf{(IV.64)}$$

Ensuite, la propagation des sorties de la couche d'entré, à partir de la couche cachée vers la couche de sortie, est donné par:

$$Y_k = f\left(\sum_{j=1}^{l} Y_j w_j + b_j\right) \quad \textbf{(IV.65)}$$

Enfin, les sorties sont les suivantes:

$$O_k = f\left(\sum_{k=1}^{n} Y_k w_k + b_k\right) \quad \textbf{(IV.66)}$$

Où f est la fonction d'activation (tangente hyperbolique) utilisée dans le réseau neuronal à trois couches, définie par:

$$f(x) = \frac{2}{1+e^{-2x}} - 1 \quad \textbf{(IV.67)}$$

Au niveau de la couche de sortie, l'erreur entre la sortie désirée T_k et la sortie réelle O_k est calculée par:

$$E_k = \left(1 - O_k^{\,2}\right)\left(T_k - O_k\right) \quad \textbf{(IV.68)}$$

L'erreur calculée est propagée vers la couche cachée à l'aide de la formule suivante:

$$E_j = \left(1 - Y_k^{\,2}\right)\sum_{k=1}^{n} w_k . E_k \quad \textbf{(IV.69)}$$

Ensuite, la propagation en arrière de l'erreur calculée à partir de la couche cachée à la couche d'entrée est la suivante:

$$E_i = \left(1 - Y_j^{\,2}\right)\sum_{j=1}^{l} w_j . E_j \quad \textbf{(IV.70)}$$

Les biais et les poids des connexions de la couche d'entrée i, de la couche cachée j et la couche de sortie k sont ajustés par:

$$\begin{cases} \Delta w_p = Y_p . E_p \\ \Delta b_p = E_p \end{cases} \quad pour \quad p = i, j, k \qquad \textbf{(IV.71)}$$

Ainsi à chaque itération d'apprentissage, les biais et les poids de connexions sont mis à jour tant que l'erreur d'apprentissage n'est pas inférieure à un seuil fixé auparavant, ou tant que l'algorithme n'a pas dépassé un certain nombre d'itérations.

IV.5.3 Séparateurs à Vaste Marge (SVM)

IV.5.3.1 Historique

Le séparateur à vastes marges (SVM, de l'anglais Support Vector Machine ou encore la Machine à Vecteurs de Support en français) est un algorithme d'apprentissage supervisé qui produit un classificateur de données. Les séparateurs à vastes marges sont des classificateurs qui reposent sur deux idées clés, qui permettent de reformuler le problème de classement comme un problème d'optimisation quadratique. La première idée clé est la notion de marge maximale qui n'est que la distance maximale entre la frontière de séparation et les échantillons les plus proches. Ces derniers sont appelés vecteurs supports. D'où le nom de la méthode. La deuxième idée clé des SVM est de transformer l'espace de représentation des données d'entrées qui ne sont pas linéairement séparables en un espace de plus grande dimension, dans lequel il est probable qu'il existe un séparateur linéaire. L'idée des hyperplans à marge maximale a été explorée dès 1963 par Vladimir Vapnik et A. Lerner [***186***], et en 1973 par Richard Duda et Peter Hart dans leur livre Pattern Classification [***170***]. L'utilité des fonctions noyaux dans le contexte de l'apprentissage artificiel a été montrée dès 1964 par Aizermann, Bravermann et Rozoener [***187***]. Ce n'est toutefois qu'en 1992 que ces idées seront bien comprises et rassemblées par Boser, Guyon et Vapnik dans un article, qui est l'article fondateur des séparateurs à vaste marge [***188***]. À partir de 1995, qui correspond à la publication du livre de Vapnik [***189***], les SVM gagnent en popularité et sont utilisés dans de nombreuses applications.

IV.5.3.2 SVM binaire

Le classificateur SVM (SVM : Support Vector Machine en anglais) consiste à trouver le séparateur optimal entre les différentes classes à partir d'un ensemble d'exemples appelé

base d'apprentissage. Pour un SVM binaire, les données ne contiennent que deux classes seulement. Il produit un classificateur linéaire pour des données linéairement séparable. Grâce à la stratégie des noyaux, le SVM est en mesure de produire une frontière de décision complexe lorsqu'il s'agit d'un ensemble de données linéairement non séparable. Ses bonnes performances et sa rapidité d'exécution en font un algorithme très utilisé. De plus, selon le type des caractéristiques utilisées de la base d'apprentissage, on peut distinguer deux types de classificateurs SVM binaire : Le classificateur linéaire qui opère sur les données linéairement séparables et le classificateur non linéaire qui utilise des fonctions noyaux ou des transformations des données non linéairement séparables afin de les classifier.

IV.5.3.2.1) SVM linéaire

Un classificateur SVM linéaire est défini par un hyperplan qui sépare en deux classe l'espace d'entrée des exemples $X \subseteq IR^n$ et qui agit en tant que frontière de décision : les exemples situés d'un côté de l'hyperplan sont classés positivement et ceux situés de l'autre côté sont classés négativement. On caractérise cet hyperplan par un vecteur $w \subseteq IR^n$ perpendiculaire à l'hyperplan et une valeur de biais $b \subseteq IR$. L'équation $w \,.\, x + b$ définit l'hyperplan dans l'espace des exemples. La valeur de sortie du classificateur correspondant est donnée par :

$$h(x) = signe(w \,.\, x - b) = signe\left(\sum_{i=1}^{n} w_i \, x_i - b\right) \qquad \textbf{(IV.72)}$$

Cette première version de l'algorithme appelé aussi SVM binaire à marge rigide, qui s'applique seulement aux ensembles d'entraînement linéairement séparables. Dans ce contexte, il existe différents classificateurs linéaires ayant un risque empirique nul. Le classificateur SVM à marge rigide trouve l'hyperplan dont la marge géométrique est maximale :

$$\gamma = \max_{w,\,b}\left[\min_{i} \frac{y_i\left[w \cdot \varphi(x_i) - b\right]}{\|w\|}\right] \qquad \textbf{(IV.73)}$$

La Figure IV-17 suivante montre les frontières de décision d'un SVM à marge rigide. Les trois exemples encerclés correspondent aux vecteurs de support.

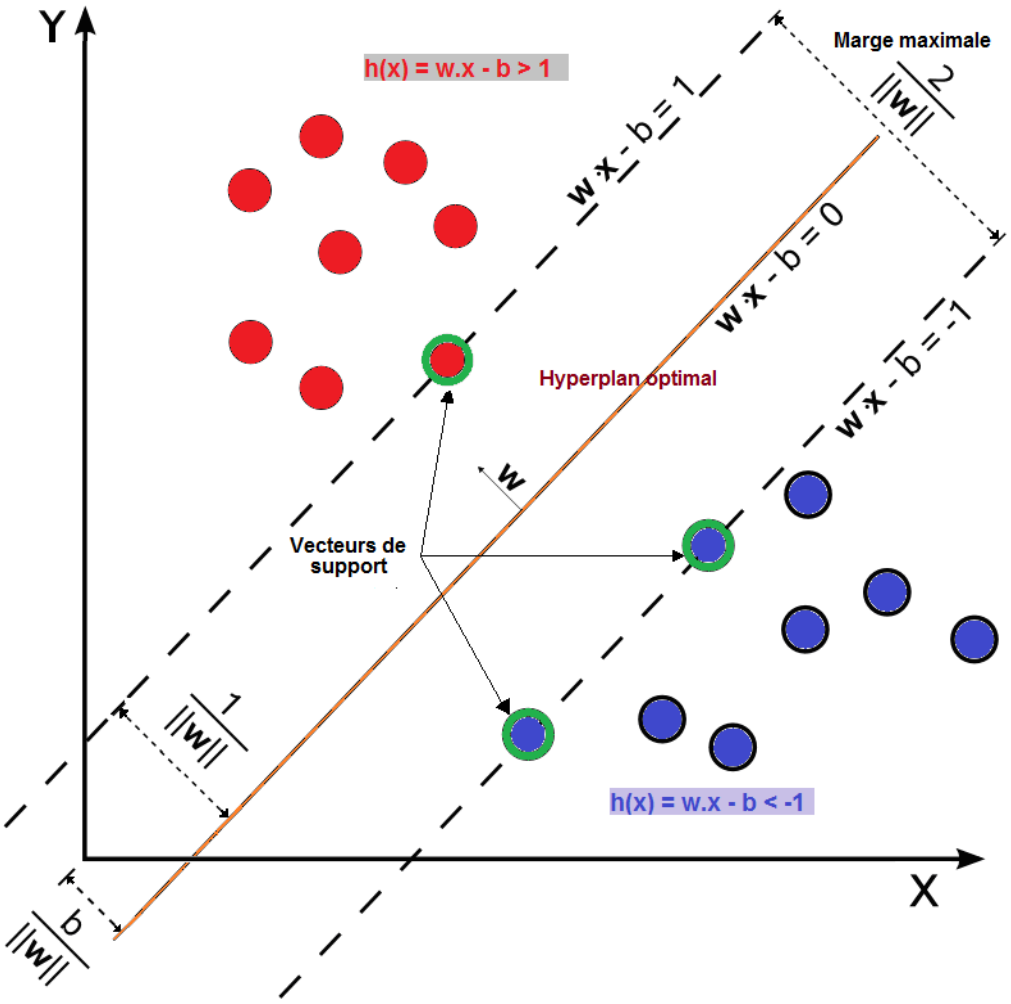


Figure IV-17 : Frontière de décision d'un SVM à marge rigide.

Les exemples qui ont la marge maximale sont appelés vecteurs de support. Ces exemples définissent deux hyperplans, parallèles à la frontière de séparation, entre lesquels n'est situé aucun exemple (voir la Figure IV-17). Alors que plusieurs vecteurs de même direction définissent la frontière de séparation recherchée, le SVM choisit l'unique séparateur de marge maximale.

Pour un ensemble d'entraînement donné, il existe une multitude de classificateurs linéaires possédant le même risque empirique. Afin de différencier ces classificateurs entre eux, on s'intéresse notamment à la distance entre la frontière de décision et les exemples de l'ensemble d'entraînement. Les SVM consistent alors à trouver l'Hyperplan Séparateur Optimal qui maximise la distance entre l'hyperplan et les deux classes [***190***.]. Cette distance est appelée marge. Il est préférable qu'elle soit maximale. Il faut noter qu'un classificateur linéaire classifie correctement un exemple lorsque sa marge est positive et incorrectement lorsque sa marge est négative.

Intuitivement, le fait d'avoir une marge plus large procure plus de sécurité lors de la classification d'un nouvel exemple. De plus, si un classificateur se comporte le mieux vis-à-vis des données d'apprentissage, il est clair qu'il sera aussi celui qui permettra au mieux de classer les nouveaux exemples. Dans la figure qui suit, la partie droite montre qu'avec un hyperplan optimal, un nouvel exemple reste bien classé alors qu'il tombe dans la marge. On constate sur la partie gauche qu'avec une plus petite marge, l'exemple se voit mal classé.

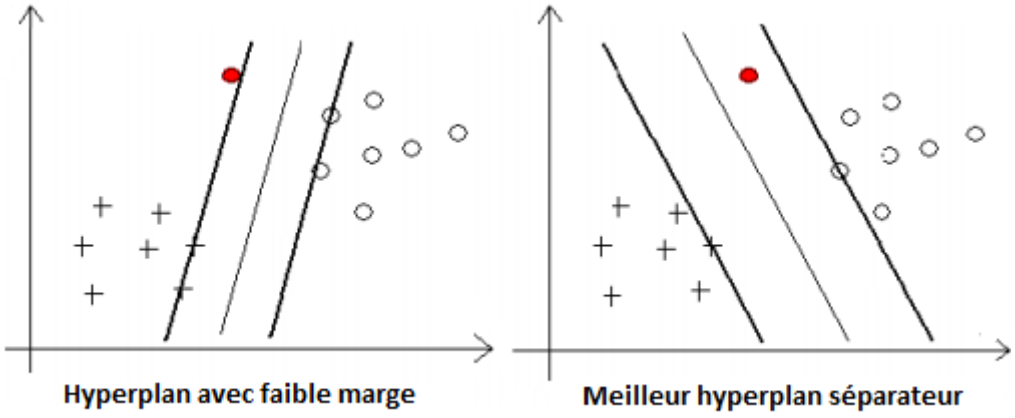


Figure IV-18 : Classification d'un nouvel exemple par un SVM à marge faible contre un SVM à marge optimale.

Une donnée définie par le couple (x, y) est bien classée si et seulement si :

$$y\,h(x) = y(w\cdot x - b) \succ 0 \qquad \textbf{(IV.74)}$$

Du fait que le couple (w, b) est défini à un coefficient multiplicatif près, on s'impose alors:

$$y\,h(x) = y(w\cdot x - b) \succ 1 \qquad \textbf{(IV.75)}$$

Le problème que résoud le SVM linéaire à marge rigide est réduit à un problème de programmation quadratique :

$$\begin{aligned} &\textit{Minimiser}: && \frac{1}{2}\|w\|^2 \\ &\textit{Sous les contraintes}: && y_i\left[w\cdot\varphi(x_i) - b\right] \geq 1 \quad \forall i \in \{1, \cdots, m\} \end{aligned} \qquad \textbf{(IV.76)}$$

Cette optimisation qui permet de choisir le meilleur hyperplan séparateur est appelée problème primal. Il est en effet plus aisé de minimiser $\|w\|^2$ plutôt que $\|w\|$. Il est possible de

résoudre ce problème efficacement à l'aide de la méthode des multiplicateurs de Lagrange, tel que décrit dans [***188***.]. Lorsqu'on introduit les multiplicateurs de Lagrange, le problème d'optimisation est dit dual.

Un ensemble d'entraînement n'est alors linéairement séparable que lorsqu'il existe un hyperplan optimal qui classifie correctement tous ses exemples. La marge d'un classificateur sur cet ensemble désigne la marge de l'exemple situé le plus près de la frontière de décision.

IV.5.3.2.2) SVM non linéaire

Il existe plusieurs ensembles de données sur lesquels aucun classificateur linéaire ne possède un risque empirique nul. Autrement dit, ces ensembles ne sont pas linéairement séparables. Dans ce contexte, les classificateurs linéaires, tels qu'exprimés jusqu'à maintenant, peuvent s'avérer peu performants. Ainsi, il est parfois utile d'exprimer des classificateurs par des frontières de décision plus complexes qu'un simple hyperplan [***188***.]. En plus de la notion des noyaux, la notion de projection et transposition des données vers un autre espace plus dimensionnel sont évoqués pour traiter ce problème.

a) Fonction de projection et transposition des données

Une première façon de représenter ces séparateurs complexes est de transposer les données de l'espace d'entrée $X \subseteq IR^n$ vers un espace de dimension plus élevée $X' \subseteq IR^{n'}$ (avec $n \prec n'$). La Figure IV-19 suivante illustre la transformation des données linéairement non séparables en un ensemble de données linéairement séparables et qui soient facile à traiter.

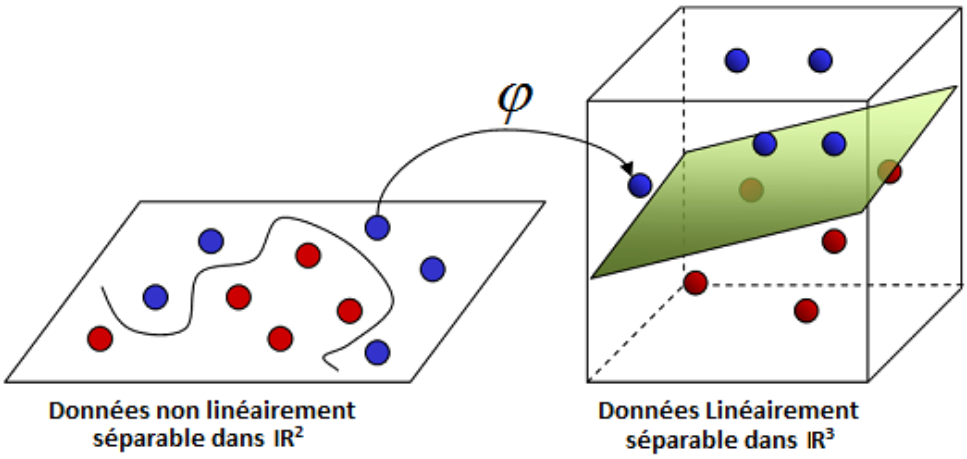


Figure IV-19 : Transformation des données linéairement non séparables en un ensemble de données linéairement séparables.

Par cette stratégie, chaque exemple $x \in X$ est transformé par une fonction en un vecteur de caractéristiques $\varphi(x) = (\varphi_1(x), \varphi_2(x), \cdots, \varphi_{n'}(x)) \in X'$

Le vecteur w de dimension n' caractérise un séparateur linéaire dans l'espace X' donné par :

$$h(x) = signe(w.\varphi(x) - b) = signe\left(\sum_{i=1}^{n'} w_i \ \varphi_i(x) - b\right) \quad \textbf{(IV.77)}$$

Ainsi, le classificateur résultant exprime un séparateur non-linéaire dans l'espace des exemples X. Pour décider de la classe d'un vecteur x, $\varphi(x)$ pourrait alors être calculé, et passé à son tour comme argument du séparateur pour en connaitre la classe. En fait, il n'est pas astucieux de procéder de la sorte, il serait plutôt préférable d'éviter le calcul explicite de la fonction $\varphi(x)$ en remarquant que le problème d'optimisation ne fait intervenir les vecteurs que via leurs produits scalaires. Ceci justifie l'utilisation de la notion des fonctions noyaux.

b) Fonctions Noyaux

Le fait de représenter une frontière de décision complexe par un séparateur linéaire dans un espace de caractéristiques de très haute dimensionnalité peut engendrer un coût de calcul élevé. Fréquemment, l'étude des algorithmes d'apprentissage révèle que les seules opérations manipulant les vecteurs de caractéristiques consistent en des produits scalaires. La stratégie du noyau permet alors de substituer ces produits scalaires par une fonction $k(x, x')$ rapidement calculable. Le résultat de $k(x, x')$ doit être équivalent au produit scalaire entre $\varphi(x)$ et $\varphi(x')$ dans un certain espace de caractéristiques X'. Ainsi, le noyau k est associé à φ par :

$$k(x, x') = \varphi(x).\varphi(x') = \sum_{i=1}^{n'} \varphi_i(x)\,\varphi_i(x') \quad \textbf{(IV.78)}$$

Il est possible de représenter implicitement le vecteur w par une combinaison linéaire des vecteurs correspondant aux exemples d'entraînement $\{y_1 \varphi(x_1), y_2 \varphi(x_2), \cdots, y_m \varphi(x_m)\}$ dans l'espace des caractéristiques. Le vecteur $\alpha \in IR^m$ contient les poids associés à cette combinaison linéaire :

$$w = \sum_{i=1}^{m} y_i \, \alpha_i \, \varphi(x_i) \qquad \textbf{(IV.79)}$$

Dans ce cas, on peut également recourir au noyau lors de la classification d'un exemple x :

$$h(x) = signe(w.\varphi(x) - b) = signe\left(\sum_{i=1}^{m} y_i \, \alpha_i \, \varphi(x_i).\varphi(x) - b \right)$$
$$= signe\left(\sum_{i=1}^{m} y_i \, \alpha_i \, k(x, x_i) - b \right) \qquad \textbf{(IV.80)}$$

Il y a des conditions mathématiques, appelées théorème de Mercer [***191***], qui permettent de dire si une fonction est un noyau ou non, sans construire la projection dans l'espace des caractéristiques.

En fait, il faut assurer que pour tout ensemble d'exemples de longueur m, la matrice carré de Gram $K_{ij} = k(x_i, x_j) = k(x_j, x_i)$ associée à un noyau k soit définie positive et symétrique. Sous cette condition, le noyau positif de Mercer définit donc bien un certain espace de Hilbert où s'exerce le produit scalaire entre les données. Il existe deux façons de construire un noyau positif de Mercer :

1. soit en s'appuyant sur une transformation $\varphi(x)$de X sur un espace muni d'un produit scalaire et l'on défini le noyau à travers ce produit scalaire.
2. soit en utilisant les propriétés algébriques des noyaux positifs :
   - un noyau séparable est un noyau positif,
   - la somme de deux noyaux positifs est un noyau positif,
   - le produit de deux noyaux positifs est un noyau positif,
   - le produit tensoriel de deux noyaux positifs est un noyau positif,
   - le passage à la limite conserve la positivité : si la limite d'une suite de noyaux positifs existe, c'est aussi un noyau positif.

Les noyaux positifs se divisent en deux grandes familles principales : les noyaux radiaux qui dépendent d'une distance et les noyaux projectifs qui sont définis à partir d'un produit scalaire.

Trois types de noyaux sont couramment utilisés. Le noyau linéaire est un simple produit scalaire définit par:

$$k(x, x') = x \cdot x' \tag{IV.81}$$

Le noyau polynomial permet de représenter des frontières de décision par des polynômes de degré p définit par:

$$k(x, x') = (x \cdot x' + 1)^p \tag{IV.82}$$

Le noyau gaussien ou RBF (de l'anglais Radial Basis Function) est un classificateur davantage complexe qui détermine la frontière de décision en effectuant une somme pondérée de fonctions gaussiennes centrées sur les exemples d'entraînement :

$$k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right) \tag{IV.83}$$

Les classificateurs exprimés par un noyau linéaire sont équivalents aux séparateurs linéaires exprimés directement dans l'espace des exemples (sans utiliser la stratégie du noyau). Aussi, les variables p et σ du noyau polynomial et du noyau RBF constituent des hyper-paramètres des algorithmes d'apprentissage. Leur valeur modifie grandement le type de frontières de décision générées et, conséquemment, la qualité des classificateurs résultants. La Figure IV-20 suivante indique les frontières de décision générées sur un ensemble de données à deux dimensions à l'aide du Noyau linéaire, le Noyau polynomial de degré 2 et le Noyau RBF.

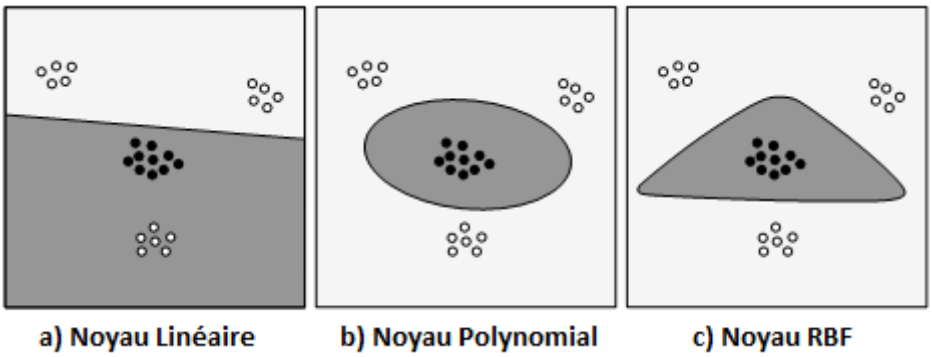


Figure IV-20 : Frontières de décision générées sur un ensemble de données à deux dimensions à l'aide du Noyau linéaire, du Noyau polynomial de degré 2 et du Noyau RBF.

Le SVM binaire non linéaire à marge floue [***192***] généralise le problème afin de permettre certaines erreurs de classification, ce qui permet d'utiliser l'algorithme sur des ensembles d'entraînement qui ne sont pas linéairement séparables. Pour ce faire, on associe à chaque exemple d'entraînement une variable d'écart ξ_i définie par :

$$\xi_i = \max\left(0, 1 - y_i \left[w \cdot \varphi(x_i) - b\right]\right) \quad \textbf{(IV.84)}$$

La Figure IV-21 suivante montre les exemples de variables écarts entre le plan séparateur et les exemples d'entraînement mal classifiés. La somme des variables d'écart $\sum_{i=1}^{m} \xi_i$ fournit une borne supérieure pour le risque empirique du classificateur. La valeur de chaque ξ_i s'interprète ainsi :

- $\xi_i \succ 1$: L'exemple x_i est mal classifié.
- $0 \prec \xi_i \prec 1$: x_i est bien classifié, mais il est situé à l'intérieur de la marge du classificateur SVM.
- $\xi_i = 0$: x_i est bien classifié et il est situé à l'extérieur de la marge du classificateur SVM.

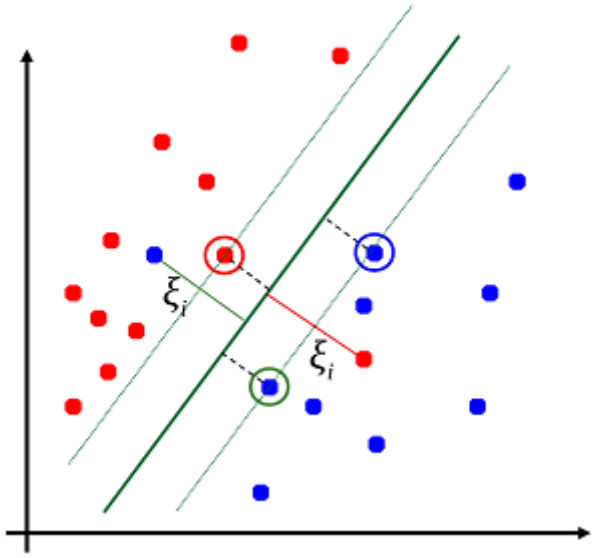


Figure IV-21 : Exemples d'écarts entre le plan séparateur et les exemples mal classifiés.

Lors de l'apprentissage, le classificateur SVM à marge floue effectue un compromis entre la marge géométrique et la pénalité due aux variables d'écart. L'ampleur de ce compromis est dictée par un hyper-paramètre de l'algorithme, le paramètre de marge floue,

désigné par la constante C. L'objectif du classificateur SVM à marge floue est de minimiser la fonction suivante :

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_i \qquad \textbf{(IV.85)}$$

Une petite valeur de C incite à tolérer les erreurs afin d'accentuer la marge. Lorsque la valeur de C est très grande, l'algorithme se comporte comme un SVM à marge rigide et ne tolère aucune erreur. En pratique, on doit souvent exécuter l'algorithme avec une variété de valeurs de C afin de trouver celle qui convient le mieux aux données étudiées.

Le problème que se pose le classificateur SVM à marge floue s'énonce aussi sous la forme d'un problème quadratique :

$$\begin{aligned} &\textit{Minimiser}: && \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_i \\ &\textit{Sous les contraintes}: && \begin{cases} y_i\left[w\cdot\varphi(x_i)-b\right] \geq 1-\xi_i \\ \xi_i \geq 0 \end{cases} \qquad \forall i \in \{1,\cdots,m\} \end{aligned} \qquad \textbf{(IV.86)}$$

Tel que démontré dans [***192***], il est important de remarquer qu'il s'agit d'un problème primal d'optimisation convexe qui peut être résolu en introduisant les multiplicateurs de Lagrange pour passer au problème dual. Par conséquent, sa solution est unique. Plusieurs algorithmes permettent de résoudre ce problème efficacement [***193***].

Jusqu'à présent, nous n'avons présentés que des classificateurs SVM binaire à deux classes. Cette notion doit être étendue aux classificateurs multi classes.

IV.5.3.3 SVM Multi classes

Les SVM ont été initialement conçus pour la classification binaire. Plusieurs méthodes ont été proposées pour construire un classificateur SVM multi-classes en combinant plusieurs classificateurs binaires. Les ensembles de données à classer peuvent être linéairement séparables ou non linéairement séparables. Les cas non linéairement séparables nécessitent l'utilisation d'une fonction noyau et d'une fonction de projection afin d'obtenir des ensembles de données linéairement séparables [***194***] [***195***].

Dans bien des cas, il s'agit de construire un noyau adapté et spécifique aux données à traiter. Les classificateurs binaires utilisés dans notre cas sont basées sur la fonction du noyau gaussienne définie par:

$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2} \right) \quad \textbf{(IV.87)}$$

Avec $\sigma = 1$.

D'autres fonctions peuvent être utilisées comme fonction de noyau pour chaque classificateur binaire.

Les Séparateurs à vaste marge ont été développés pour traiter des problèmes binaires mais ils peuvent être adaptés pour traiter les problèmes multi-classes. Dans cette thèse, les classificateurs binaires basés sur la stratégie un-contre-un et la stratégie un-contre-tous sont utilisés. La Figure IV-22 et la Figure IV-23 montrent respectivement leurs structures fondées sur les classificateurs SVM binaires.

IV.5.3.3.1) Classificateur SVM un-contre-tous

L'idée consiste tout simplement à transformer le problème à N classes en N classificateurs binaires. Le classificateur SVM un-contre-tous contient N classificateur binaire, où N est le nombre de classes dans l'ensemble des données. L'ième SVM binaire est entrainé avec tous les exemples de données dans l'i-ième classe N°i avec des étiquettes positives, et tous les autres exemples de données avec des étiquettes négatives. La classification est donnée par le classificateur qui répond le mieux.

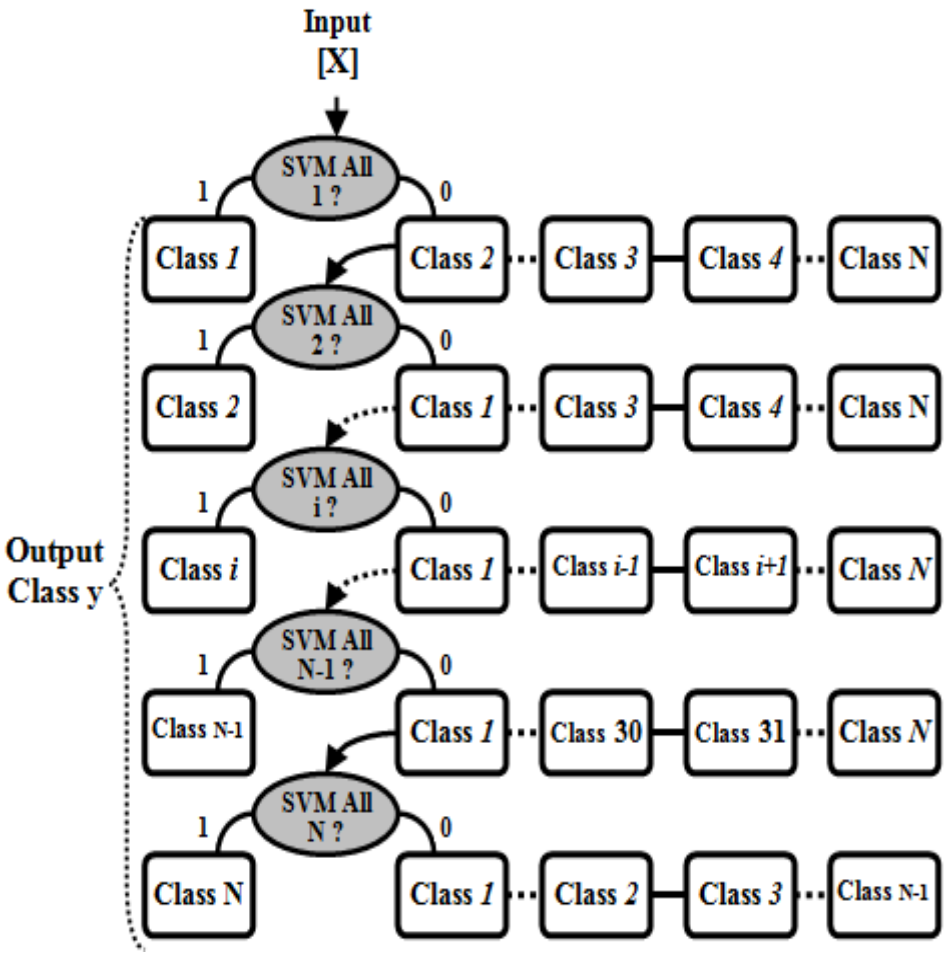


Figure IV-22 : Structure du classificateur SVM multi-classes un-contre-tous.

Pour construire le modèle de SVM multi-classes basé sur la stratégie un-contre-tous à partir des classificateurs binaires, les classes sont divisées en deux groupes: le premier groupe est formé par une classe, et le second groupe est constitué de toutes les autres classes. Le classificateur SVM obtenu est entrainé pour décider si la classe appartient au premier groupe ou au deuxième groupe de classes. Ce processus est répété pour le deuxième groupe qui contient plus de deux classes jusqu'à ce qu'il n'y aura qu'une seule classe pour chaque groupe. Le processus doit s'arrêter là, c'est le critère d'arrêt du processus de formation et entrainement du classificateur SVM multi-classes. Ainsi, en suivant cette manière, le classificateur SVM multi-classes est transformé en plusieurs classificateurs binaires SVM (Figure IV-22). Chaque classificateur SVM binaire est entrainé en utilisant une matrice de données d'apprentissage, où chaque rangée correspond à des caractéristiques extraites comme une observation à partir d'une classe. Après la phase d'entrainement, le modèle SVM multi-classes est en mesure de décider de la bonne classe pour un vecteur de caractéristiques d'entrée. Pour classifier un objet, ce vecteur de caractéristiques d'entrée est présenté de

manière itérative au j'ième classificateur binaire (N°j) du premier à l'Nième classificateur tant que le résultat est négatif. Lorsque l'ième classificateur binaire donne un résultat positif, le processus est arrêté. Cela signifie que l'objet appartient à l'ième classe.

Soit $X = (X_1, X_2, \cdots, X_N)$ les N vecteurs caractéristiques comprenant les données d'apprentissage, X_k un exemple de vecteur d'entrée et $C = (C_1, C_2, \cdots, C_N)$ les N classes correspondantes. $Svm\_All = \{Svm_1, Svm_2, \cdots, Svm_N\}$ dénote les N SVM binaires à construire et entrainer. L'Algorithme IV.7 d'apprentissage et de classification permettant de traduire le principe de la Stratégie un-contre-tous des SVM Multi-Classes décrit précédemment est le suivant :

Algorithme IV.7 : SVM multi-classes basé sur la Stratégie un-contre-tous.

Algorithme : SVM multi-classes basé sur la Stratégie un-contre-tous.

Entrées : $X = (X_1, X_2, \cdots, X_N)$ les N vecteurs caractéristiques d'apprentissage, X_k un exemple de vecteur d'entré ;

$C = (C_1, C_2, \cdots, C_N)$ les N classes correspondantes.

Sorties : $Svm\_All = \{Svm_1, Svm_2, \cdots, Svm_N\}$ Les N SVM binaire un contre tous à entrainer,

Classe la classe correspondante à X_k .

Début

Fonction Apprentissage_Svm_All(X, C)

$i \leftarrow 1$;

Tant Que $(i \le N)$ **Faire**

$T = (T_1, T_2, \cdots, T_N) \leftarrow 0$;

Pour j=1 à N **Faire**

Si $(C_i = C_j)$ **Alors**

$T_j \leftarrow 1$;

Fin Si ;

Fin Pour ;

$Svm_i \leftarrow Entrainer\_Svm\_Binaire(Svm_i, X, T)$;

$i \leftarrow i + 1$;

Fin Tant Que ;

Retourner $Svm\_All = \{Svm_1, Svm_2, \cdots, Svm_N\}$;

Fonction Classification_Svm_All(X_k)

$i \leftarrow 1$;

Répéter

$S \leftarrow Classer\_Svm\_Binaire(Svm_i, X_k)$;

Si $(S = 1)$ **Alors**

$Y \leftarrow C_i$;

Fin Si ;

$i \leftarrow i + 1$;

Tant Que $(i \leq N \quad et \quad S = 0)$;

Retourner Y **;**

Fin.

IV.5.3.3.2) Classificateur SVM un-contre-un

Une autre méthode importante est appelée SVM un-contre-un. À partir de N classes dans les ensembles de données, cette méthode transforme le problème en $N(N-1)/2$ classificateurs binaires où chacun est entraîné sur les données de deux classes. Chaque classe i est en effet comparée à chacune des autres classe j. La classification est donnée par le vote majoritaire.

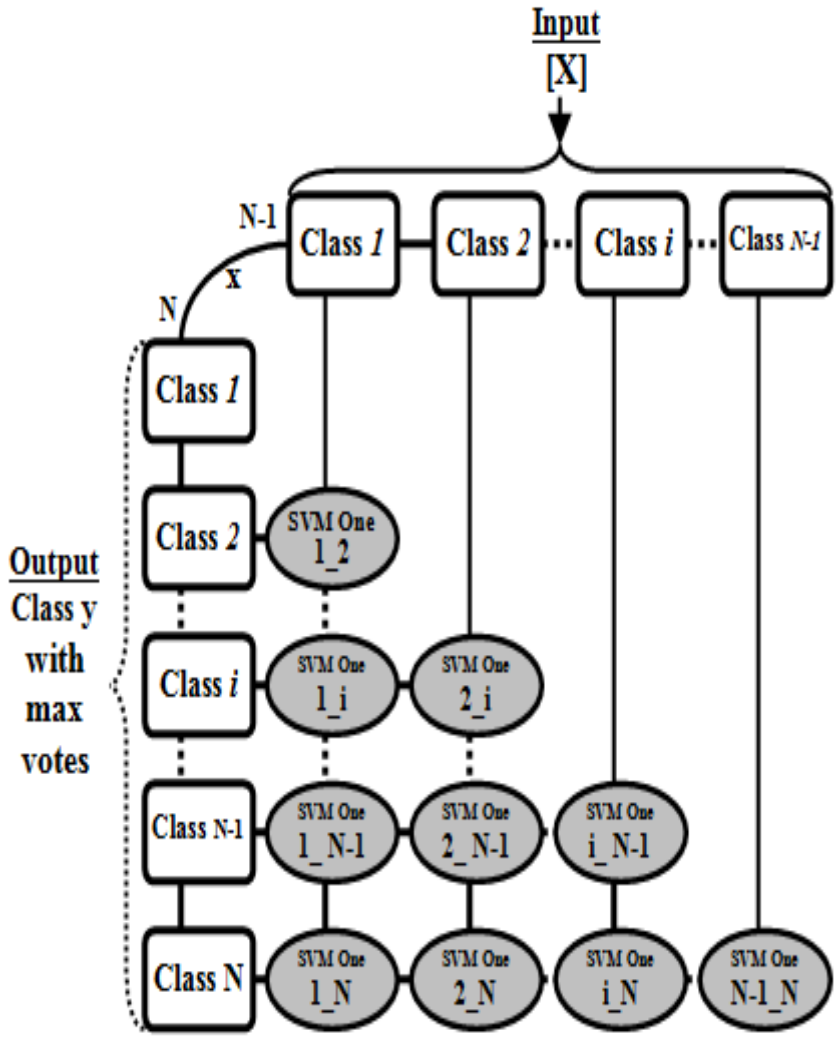


Figure IV-23 : Structure du classificateur SVM multi-classes un-contre-un.

Pour concevoir et étendre le classificateur SVM binaire en un classificateur SVM multi-classes un-contre-un, deux groupes d'exemples de données sont construits à partir de deux classes. Le classificateur SVM binaire obtenu est entraîné à décider s'il s'agit de la première classe ou de la deuxième classe. Ce processus est répété pour un autre couple de classes jusqu'à terminer de balayer tous les couples possibles des classes à partir de l'ensemble des données. Ainsi, en suivant cette manière, le classificateur SVM multi-classes est transformé en $N(N-1)/2$ classificateur SVM binaires (Figure IV-23). Chaque classificateur SVM binaire est entraîné en utilisant une matrice de données d'apprentissage, où chaque rangée correspond à des caractéristiques extraites en tant qu'observation à partir d'une classe. Pour la classification d'un objet avec un vecteur de caractéristiques d'entrée, chaque classificateur SVM binaire du modèle multi-classes SVM un-contre-un décide et vote

pour une seule classe. La classe ayant la majorité des votes est supposée être la bonne classe à qui appartient l'objet.

Supposons qu'on a $X = (X_1, X_2, \cdots, X_N)$ les N vecteurs caractéristiques comprenant les données d'apprentissage, X_k un exemple de vecteur d'entrée et $C = (C_1, C_2, \cdots, C_N)$ les N classes correspondantes. $Svm\_One = \{Svm_{i,j} \quad \forall i, j \in \{1, \cdots, N\} et\ i \prec j\}$ dénote les $N(N-1)/2$ SVM binaires à construire et entrainer. De même pour le modèle un-contre-tous, l'Algorithme IV.8 d'apprentissage et de classification permettant de traduire le principe de la Stratégie un-contre-un des SVM Multi-Classes est le suivant :

Algorithme IV.8 : SVM multi-classes basé sur la Stratégie un-contre-un.

Algorithme : SVM multi-classes basé sur la Stratégie un-contre-un.

Entrées : $X = (X_1, X_2, \cdots, X_N)$ les N vecteurs caractéristiques d'apprentissage, X_k un exemple de vecteur d'entré ;

$C = (C_1, C_2, \cdots, C_N)$ les N classes correspondantes.

Sorties : $Svm\_One = \{Svm_{i,j} \quad \forall i, j \in \{1, \cdots, N\} et\ i \prec j\}$ Les $N(N-1)/2$ SVM binaire un contre un à entrainer,

Y est la classe correspondante à X_k .

Début

Fonction Apprentissage_Svm_One(X, C)

Pour i=1 à N-1 **Faire**

Pour j=i+1 à N **Faire**

$D \leftarrow \{X_i, X_j\}$;

$T \leftarrow \{C_i, C_j\}$;

$Svm_{i,j} \leftarrow Entrainer\_Svm\_Binaire(Svm_{i,j}, D, T)$;

Fin Pour ;

Fin Pour ;

Retourner $Svm\_One = \{Svm_{i,j} \quad \forall i, j \in \{1, \cdots, N\} et\ i \prec j\}$;

Fonction Classification_Svm_One(X_k)

$T = (T_1, T_2, \cdots, T_N) \leftarrow 0$; *// compteurs de votes*

Pour i=1 à N-1 **Faire**

Pour j=i+1 à N **Faire**

$$S \leftarrow Classer\_Svm\_Binaire\left(Svm_{i,j}, X_k\right);$$

$$T_S \leftarrow T_S + 1;$$

Fin Pour ;

Fin Pour ;

// Choisir la classe ayant le maximum de votes.

$$Y \leftarrow \left\{ C_l \quad / \quad l = \arg\max_{S \in \{1,\cdots,N\}} \{T_S\} \right\}$$

Retourner Y **;**

Fin.

IV.5.4 Réseaux bayésiens

Dans cette partie, nous présentons une méthode de classification reposant sur une approche probabiliste basée sur la règle de Bayes. L'intérêt de cette approche est qu'elle permet d'intégrer des connaissances a priori, ce que ne permettent pas la plupart des méthodes de classification.

Il est important de noter que dans l'approche bayésienne, on mesure la probabilité d'occurrence d'un événement si un autre événement est vérifié : Le deuxième événement joue le rôle d'une hypothèse préliminaire que l'on suppose remplie pour estimer la probabilité d'occurrence de l'événement recherché. L'approche bayésienne est basée alors sur la probabilité conditionnelle qui estime la probabilité d'occurrence d'un événement en supposant qu'un autre événement est vérifié. Dans la probabilité conditionnelle, le fait de savoir si l'événement a réellement lieu n'est pas intéressant; la probabilité dans cette situation est estimée en supposant que l'événement a lieu.

IV.5.4.1 Historique

Les réseaux bayésiens sont le résultat d'une convergence entre la discipline de l'intelligence artificielle et la discipline des statistiques et probabilités. Ils constituent aujourd'hui l'un des formalismes les plus complets et les plus cohérents pour l'acquisition, la représentation et l'utilisation de connaissances par des ordinateurs. Encore du domaine de la recherche au début des années 1990, cette technologie connaît de plus en plus d'applications,

depuis le contrôle de véhicules autonomes à la modélisation des risques opérationnels, en passant par le data mining ou la localisation des gènes.

Les réseaux bayésiens, qui doivent leur nom aux travaux de Thomas Bayes au XVIIIe siècle sur la théorie des probabilités [***196***], sont le résultat de recherches effectuées dans les années 1980, dues à J. Pearl [***197***], [***198***], [***199***], [***200***], [***201***], [***202***], [***203***] à l'Université de California, Los Angeles (UCLA). Microsoft a proposé dès 1994 un assistant de dépannage pour les problèmes d'impression dans Windows 95. Leur programme commence par proposer la solution qui paraît la plus probable pour résoudre le problème détecté.

L'objectif initial de ces travaux était d'intégrer la notion d'incertitude dans les systèmes experts. Les chercheurs se sont rapidement aperçus que la construction d'un système expert nécessitait presque toujours la prise en compte de l'incertitude dans le raisonnement.

En effet, dans la plupart des domaines complexes, un expert humain est capable de porter un jugement sur une situation, même en l'absence de toutes les données nécessaires. En médecine, par exemple, une même combinaison de symptômes peut être observée dans différentes pathologies.

Il n'y a donc pas de règle stricte qui permette de passer systématiquement d'un ensemble d'observations à un diagnostic. De plus, les informations pertinentes ne sont pas toujours observables. Pour que des systèmes experts puissent être utilisés dans de tels domaines, il faut donc qu'ils soient capables de raisonner sur des faits et des règles incertains. Dans le cadre des systèmes experts, les réseaux bayésiens constituent une approche possible pour intégrer l'incertitude dans le raisonnement. D'autres méthodes existent, mais les réseaux bayésiens présentent l'avantage d'être une approche quantitative et générative.

La structure graphique d'un réseau bayésien traduit les relations d'indépendance conditionnelle entre variables aléatoires à l'aide de liens munissant l'ensemble des ces variables de la structure de graphe orienté sans circuits; elle améliore la compréhension du modèle du fait que les différentes relations de dépendance entre les variables aléatoires en jeu dans le modèle peuvent être lues directement à partir de cette structure. Initialement c'est cette structure graphique qui a rendu les réseaux bayésiens très attractifs car elle pouvait coder la structure du domaine à modéliser. La facilité de compréhension des réseaux bayésiens est l'un des plus grands avantages des représentations qu'ils permettent, comparées par exemple à celles obtenues par les réseaux neuronaux qui sont plus difficiles à comprendre.

La correspondance qui existe entre la structure graphique et la structure probabiliste associée va permettre de ramener pour une bonne part l'ensemble des problèmes d'inférence à des problèmes de graphes. Cependant ces problèmes restent relativement complexes et donnent lieu à de nombreuses recherches [***204***].

IV.5.4.2 Théorème de Bayes

L'inférence bayésienne est un concept très simple reposant sur la règle de Bayes traitant des probabilités conditionnelles [***205***]. Cette règle est donnée par :

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)} \qquad \textbf{(IV.88)}$$

La notation précédente permet de mettre en avant H en tant qu'hypothèse et E en tant qu'évidence ou observation.

H est l'ensemble des hypothèses concurrentes qui pourront être modifiées par l'observation de données qui sont alors les évidences E. L'inférence bayésienne permet de déterminer la probabilité a posteriori p(H|E) comme conséquence d'une probabilité a priori p(H) et d'une fonction de vraisemblance p(E|H).

Chaque terme de l'équation précédente est expliqué ci-après :

- La probabilité a priori p(H) est la probabilité de H avant que E ne soit observé et correspond donc aux hypothèses du modèle avant observation du comportement du système.
- La probabilité a posteriori p(H|E) est la probabilité de chaque hypothèse après observation de l'évidence.
- La fonction de vraisemblance Likelyhood(H|E) = p(E|H) est la probabilité d'observer une évidence E sous une hypothèse H. Elle dénote alors la compatibilité entre une évidence et une hypothèse. Sa détermination repose sur le modèle de probabilité des données observées
- p(E) est la vraisemblance marginale ou encore l'évidence du modèle. Elle est la même quelle que soit l'hypothèse H.

En résumé, la probabilité a posteriori d'une hypothèse est déterminée par une vraisemblance intrinsèque, dénotée par la probabilité a priori, et une compatibilité entre l'évidence et l'hypothèse déterminée par la fonction de vraisemblance.

IV.5.4.3 Définition formelle d'un réseau bayésien

Un réseau bayésien $B = (G, \theta)$ est un modèle graphique probabiliste représentant des variables aléatoires sous la forme d'un graphe orienté acyclique. Il est défini par :

- $G = (X, E)$, où X est l'ensemble des nœuds (ou sommets) et où E est l'ensemble des arcs, G est un graphe (un arbre) dirigé sans circuit (DAG pour Directed Acyclic Graphe en anglais) dont les sommets sont associés à un ensemble de variables aléatoires $X = \{X_1, X_2, \cdots, X_n\}$,
- $\theta = \{P(X_i | Pa(X_i))\}$, ensemble des probabilités de chaque nœud X_i conditionnellement à l'état de ses parents $Pa(X_i)$ dans G.

La partie graphique du réseau bayésien indique les dépendances ou indépendances entre les variables et donne un outil visuel de représentation des connaissances, outil plus facilement appréhendable et compréhensible par ses utilisateurs. De plus, l'utilisation de probabilités permet de prendre en compte l'incertain, en quantifiant les dépendances entre les variables.

Ainsi, les réseaux bayésiens associent une partie qualitative que sont les graphes et une partie quantitative représentant les probabilités conditionnelles associées à chaque nœud du graphe relativement aux parents [***206***]. La partie qualitative exprime des indépendances conditionnelles entre variables et des liens de causalités et ce grâce à un graphe orienté acyclique dont les nœuds correspondent à des variables aléatoires. La partie quantitative est constituée de tables de probabilités dans le cas discret ou de distribution gaussiennes dans le cas continu.

Pearl et all. [***207***] ont aussi montré que les réseaux bayésiens permettaient de représenter de manière compacte la distribution de probabilité jointe sur l'ensemble des variables :

$$P(X) = P(X_1, X_2, \cdots, X_n) = \prod_{i=1}^{n} P(X_i | Pa(X_i)) \qquad \textbf{(IV.89)}$$

Dans cette expression, $Pa(X_i)$ est l'ensemble des parents du nœud X_i dans le graphe G du réseau bayésien.

Cette décomposition d'une fonction globale en un produit de termes locaux, dépendant uniquement du nœud considéré et de ses parents dans le graphe, est une propriété fondamentale des réseaux bayésiens. Elle est à la base des premiers travaux portant sur le développement d'algorithmes d'inférence, qui calculent la probabilité de n'importe quelle variable du modèle à partir de l'observation même partielle des autres variables.

Les algorithmes des méthodes de propagation d'information dans un graphe utilisent évidemment la notion de probabilité conditionnelle qui permet de déterminer la probabilité de X_i sachant que X_j est observé, mais aussi le théorème de Bayes, qui permet de calculer, inversement, la probabilité de X_j sachant X_i, lorsque $P(X_i \mid X_j)$ est connu.

Cette probabilité jointe est en fait une expression simplifiée. La simplification a pu être obtenue grâce à la règle de Bayes [***205***], la probabilité jointe peut être décomposée de la façon suivante [***19***]:

$$\begin{aligned} P(X) &= P(X_1, X_2, \cdots, X_n) = \prod_{i=1}^{n} P(X_i \mid Pa(X_i)) \\ &= P(X_n \mid X_{n-1}, \cdots, X_1) \times P(X_{n-1} \mid X_{n-2}, \cdots, X_1) \times \cdots \times P(X_2 \mid X_1) \times P(X_1) \\ &= P(X_1) \times \prod_{i=2}^{n} P(X_i \mid X_{i-1}, \cdots, X_1) \end{aligned} \quad \textbf{(IV.90)}$$

Derrière tout réseau bayésien se cache donc une hypothèse essentielle : chaque variable est indépendante de ses non descendants étant donnés ses parents dans le graphe. Les propriétés de dépendance et d'indépendance conditionnelles d'un tel modèle sont visualisables dans son graphe.

La modélisation d'un problème par un réseau bayésien, puis l'utilisation d'algorithmes d'inférence, ont fait des réseaux bayésiens des outils idéaux pour le raisonnement ou le diagnostic à partir d'informations incomplètes.

Les réseaux Bayésiens peuvent traiter deux types de variables : discrètes ou continues. Dans le cas de variables discrètes, la somme des probabilités vaut 1. Dans le cas de variables continues, c'est l'intégrale qui vaut 1.

Chaque probabilité conditionnelle est considérée comme un paramètre du modèle. Il n'y a donc pas de distinction entre données et paramètres : les paramètres d'un modèle sont ses probabilités. Il est possible de représenter les paramètres au niveau des nœuds du graphe représentant le modèle. La représentation de ces paramètres permet d'enrichir la structure graphique du modèle en quantifiant les relations entre les variables.

IV.5.4.4 Différents modèles graphiques des réseaux bayésiens

Il existe plusieurs variantes des réseaux Bayésiens tels que [***208***] : les réseaux Bayésiens multi agents, les réseaux Bayésiens de niveaux deux, les réseaux Bayésiens orientés objets, les diagrammes d'influence, les réseaux Bayésiens dynamiques [***209***] (temporels), les réseaux Bayésiens multi entités, les filtres bayésiens [***210***] qui sont des réseaux Bayésiens dynamiques particuliers et les réseaux Bayésiens adaptés à la classification.

Ainsi, une application donnée peut être représentée par plusieurs modèles ou structures et donc différents paramètres ou probabilités, d'où la nécessité d'apprentissage de structure et de paramètre d'un réseau bayésien afin de déterminer les paramètres optimaux.

IV.5.4.5 Apprentissage de paramètres et de structure

La construction d'un réseau bayésien consiste à trouver une structure ou un graphe et estimer ses paramètres (probabilités conditionnelles). Cependant devant une très grande base de données, on ne peut pas extraire la structure adaptée à une telle quantité de données. D'où la nécessité d'un apprentissage automatique de paramètres.

IV.5.4.5.1) Apprentissage des paramètres

L'apprentissage de paramètres consiste à supposer que la structure du réseau est déjà fixée et donc à déterminer les probabilités conditionnelles de chaque variable qui se trouve dans le réseau. Les données disposées peuvent être complètes ou incomplètes, discrètes ou continues. Pour chaque cas, l'algorithme d'apprentissage des paramètres diffère. Dans le cas où toutes les variables sont observées et discrètes, la méthode la plus simple et la plus utilisée pour estimer les paramètres est l'estimation statistique de la probabilité d'un évènement par la fréquence d'apparition de l'évènement dans la base de donnée. Cette méthode est appelée maximum de vraisemblance [***211***], donnée par l'expression suivante:

$$P\left(X_i = x_k \middle| Pa\left(X_i\right) = x_j\right) = \widehat{\theta}_{i,j,k}^{MV} = \frac{N_{i,j,k}}{\sum_k N_{i,j,k}} \qquad \textbf{(IV.91)}$$

Où $N_{i,j,k}$ est le nombre d'événements dans la base de données pour lesquels la variable X_i est dans l'état x_k et ses parents $Pa\left(X_i\right)$ sont dans l'état ou la configuration x_j.

L'inconvénient de cette méthode est qu'on peut avoir une probabilité nulle à cause de l'absence d'un événement dans la base de données, ce qui est faux en réalité. Pour remédier à ce problème on fait recours à d'autres approches dites méthodes bayésiennes. Ces méthodes sont connues sous le nom de Maximum à Posteriori (MAP) et Espérance à Posteriori (EAP) [***19***]. Supposons que les paramètres x_i admettent une densité de probabilité exponentielle de Dirichlet, alors on peut écrire l'expression ci-dessous:

$$P\left(x_1, \ldots, x_n \middle| \alpha_1, \ldots, \alpha_n\right) = \frac{\Gamma\left(\sum_{i=1}^{n} \alpha_i\right)}{\prod_{i=1}^{n} \Gamma\left(\alpha_i\right)} \times \prod_{i=1}^{n} x_i^{\alpha_i - 1} \qquad \textbf{(IV.92)}$$

Avec Γ la fonction gamma d'Euler définie par : $\Gamma\left(x+1\right) = x\,\Gamma\left(x\right)$ *et* $\Gamma\left(1\right) = 1$. Et α_i sont les paramètres de la distribution de Dirichlet associée à la densité de probabilité des paramètres x_i.

Ainsi, l'approche de maximum à posteriori (MAP) est donnée par [***212***]:

$$P\left(X_i = x_k \middle| Pa\left(X_i\right) = x_j\right) = \widehat{\theta}_{i,j,k}^{MAP} = \frac{N_{i,j,k} + \alpha_{i,j,k} - 1}{\sum_k \left(N_{i,j,k} + \alpha_{i,j,k} - 1\right)} \qquad \textbf{(IV.93)}$$

Où $\alpha_{i,j,k}$ sont les paramètres de la distribution de Dirichlet associée à la loi à priori $P\left(X_i = x_k \middle| Pa\left(X_i\right) = x_j\right)$.

Une autre approche bayésienne consiste à calculer l'espérance a posteriori des paramètres $\theta_{i,j,k}$ au lieu d'en chercher le maximum. L'espérance à posteriori (EAP) est donnée par [***212***]:

$$P\left(X_i = x_k \middle| Pa\left(X_i\right) = x_j\right) = \hat{\theta}_{i,j,k}^{EAP} = \frac{N_{i,j,k} + \alpha_{i,j,k}}{\sum_k \left(N_{i,j,k} + \alpha_{i,j,k}\right)} \quad \textbf{(IV.94)}$$

Un autre estimateur appelé estimateur de Laplace [***213***], qui suppose que les paramètres de Dirichlet égales à 1, est défini par:

$$P\left(X_i = x_k \middle| Pa\left(X_i\right) = x_j\right) = \hat{\theta}_{i,j,k}^{EL} = \frac{N_{i,j,k} + 1}{\sum_k \left(N_{i,j,k} + 1\right)} \quad \textbf{(IV.95)}$$

Les estimations évoquées précédemment (maximum de vraisemblance, maximum à posteriori, espérance à posteriori et l'estimateur de Laplace) ne sont valables que si les variables sont entièrement observées. D'autres méthodes doivent donc être appliquées aux cas où certaines données sont manquantes.

Dans la plupart des travaux, les calculs des probabilités effectués reposent sur l'utilisation de variables discrètes dont les distributions de probabilité conditionnelles par rapport aux parents sont des tables. L'extension de ces résultats à des variables décrites par des distributions continues est possible. Il est également possible de mixer des variables continues et des variables discrètes comme c'est le cas dans notre application. Pour ce faire, on utilise généralement des distributions gaussiennes. La distribution d'un nœud conditionnellement à ses parents est une distribution gaussienne dont la moyenne est une combinaison linéaire de la valeur des parents et dont la variance est indépendante de la valeur des parents [***214***].

$$\begin{aligned} P\left(X_i = x_i \middle| Pa\left(X_i\right)\right) &= N\left(\mu_i + \sum_{j=1}^{n_i} \frac{\sigma_{ij}}{\sigma_j^2}\left(x_j - \mu_j\right), \sigma_i^2\right) \\ &= \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left\{-\frac{1}{2\sigma_i^2}\left(x_i - \left(\mu_i + \sum_{j=1}^{n_i} \frac{\sigma_{ij}}{\sigma_j^2}\left(x_j - \mu_j\right)\right)\right)^2\right\} \end{aligned} \quad \textbf{(IV.96)}$$

Où :

- $Pa\left(X_i\right)$ Sont les parents de X_i.
- μ_i , μ_j , σ_i *et* σ_j sont les moyennes et les variances des attributs X_i et X_j respectivement sans prendre en compte leur parents.
- n_i est le nombre de parents de X_i.

- σ_{ij} est la matrice de régression des poids.

Dans le cas de données incomplètes, Il existe plusieurs algorithmes pour estimer les données manquantes à partir des données connues, au lieu de les ignorer. Le plus utilisé est l'algorithme EM (Expectation Maximisation ou Espérance Maximisation) [***215***].

IV.5.4.5.2) Apprentissage de structure

On vient d'examiner quelques méthodes d'apprentissage des paramètres d'un réseau bayésien à partir de données complètes ou incomplètes en supposant que la structure de ce réseau était déjà connue. Se pose maintenant la question de l'apprentissage de cette structure : comment trouver la structure qui représentera le mieux un problème donné.

Une première approche consiste à rechercher les différentes indépendances conditionnelles qui existent entre les variables. Les autres approches tentent de quantifier l'adéquation d'un réseau bayésien au problème à résoudre, c'est-à dire d'associer un score à chaque réseau bayésien. Puis elles recherchent la structure qui donnera le meilleur score dans l'espace des graphes acycliques dirigés. Un parcours exhaustif est impossible en pratique en raison de la taille de l'espace de recherche qui est très grande. La formule suivante, démontrée dans [***216***], prouve que le nombre de structures possibles à partir de n nœuds est super-exponentiel (par exemple, S(5) = 29281 et S(10) = 4.2×10^{18}). Ce nombre est donné par:

$$S(n) = \sum_{i=1}^{n} (-1)^{i+1} C_n^i 2^{i(n-i)} S(n-i) \qquad pour \quad n \succ 1 \qquad \textbf{(IV.97)}$$

Où $S(n)$ le nombre de graphes possible, et n le nombre de nœuds existants.

Beaucoup de travaux se sont intéressés aux problèmes de l'apprentissage de structures [***217***]. Là aussi comme pour l'apprentissage de paramètres, on a deux cas, selon que les données sont totalement ou partiellement observables. Pour le premier cas, deux familles d'approches ont été proposées. A savoir celles qui sont basées sur des scores pour trouver la configuration optimale d'un réseau bayésien. Ces algorithmes consistent à parcourir tous les graphes possibles puis associer un score à chaque graphe. Le graphe qui possède le plus grand score va être sélectionné. Cependant, cette méthode est applicable seulement pour des problèmes de taille limitée (quelques centaines de variables). Il existe deux types de score : les scores locaux et les scores globaux [***218***]. La deuxième famille utilise des tests statistiques

afin de déterminer les indépendances entre les variables dans le réseau bayésien. On peut citer l'algorithme de « PC » pour Peter et Clark, « IC » algorithme de principe similaire, pour Inductive Causation, « BNPC » Nommée BN-PC-A et BN-PC-B pour Bayes Net Power Constructor A et B, car les mêmes auteurs ont introduit deux algorithmes, « GS » pour Greedy Search ou recherche gloutonne et la « Méthode de recherche de l'arbre de recouvrement de poids maximal » (Maximal Weight Spanning Tree ou MWST). Cette méthode s'applique à la recherche de structure d'un réseau bayésien en fixant un poids à chaque arête potentielle de l'arbre.

Comme dans l'apprentissage de paramètres, si on ne possède pas toutes les mesures qui sont issues des observations réelles, on utilise les algorithmes d'apprentissage de structure avec des données incomplètes, l'algorithme le plus utilisé est celui de EM (Expectation maximisation ou Espérance maximisation) [***219***].

En présence d'un réseau bayésien, nous pouvons extraire un certain nombre d'informations. En premier lieu, nous avons accès à la structure du réseau, celle-ci nous permet de savoir quels sont les attributs qui sont dépendants ou non, de plus nous avons accès aux probabilités conditionnelles. Le calcul de ces informations constitue l'inférence bayésienne.

IV.5.4.6 Inférence bayésienne

L'inférence bayésienne est le calcul de la probabilité de n'importe quelle variable d'un modèle probabiliste à partir de l'observation d'une ou de plusieurs autres variables. Il consiste à propager une ou plusieurs informations au sein de ce réseau, pour en déduire comment sont modifiées les croyances et les caractéristiques concernant les autres nœuds. La structure du graphe joue un rôle très important dans la complexité de ces calculs ainsi que dans le choix de la méthode d'inférence. On peut distinguer deux catégories d'algorithmes d'inférence [***220***] : l'inférence exacte et l'inférence approchée. Plusieurs méthodes ou algorithmes sont conçus spécialement pour les problèmes d'inférence exacte pour les réseaux bayésiens. On peut citer l'algorithme de passage de messages de Pearl, dont le principe est le suivant, à chaque nœud est associé un processeur qui peut envoyer des messages à ses voisins, jusqu'a ce qu'un équilibre soit atteint, en un nombre fini d'étapes. Le problème de l'algorithme de passage de messages de Pearl est qu'il ne s'applique qu'aux arbres, l'algorithme d'arbre de jonction, qui est une généralisation de l'algorithme de passage de messages de Pearl permet de faire de l'inférence sur n'importe quel type de graphe. Cette méthode est divisée en cinq étapes qui

sont : la moralisation du graphe, la triangulation du graphe, la construction de l'arbre de jonction, l'inférence dans l'arbre de jonction en utilisant l'algorithme des messages locaux et la transformation des potentiels de clique en lois conditionnelles mises à jour. A rappeler qu'une clique est un sous graphe du graphe G dont tous les nœuds sont connectés deux à deux. On peut citer encore l'algorithme d'élimination des variables ou l'algorithme d'élimination de Bucket qui consiste à marginaliser la distribution de probabilité jointe d'un réseau, en procédant variable par variable. Chaque marginalisation sur une variable X_i donne lieu à une somme des probabilités de cette variable. Parfois, cette somme vaudra 1, ce qui conduit à l'élimination de la variable X_i. On procédera alors à la marginalisation sur une des variables restantes et ainsi de suite jusqu'a ce que la distribution soit marginalisée. Le problème de cet algorithme est que l'ordre dans lequel les variables sont éliminées détermine la quantité de calcul nécessaire pour marginaliser la distribution de probabilités jointe et donc la complexité de l'algorithme. Dans le cas de la classification on peut avoir soit des variables discrètes ou bien un mélange de variables discrètes représentant les classes et les variables continues représentants les caractéristiques du modèle. Dans les deux cas, le problème de l'inférence revient à calculer les probabilités à posteriori, ainsi la classe choisie est celle qui maximise ces probabilités. Elles sont données par :

$$P(C_i|X) = \begin{cases} P(C_i)\prod_{j=1}^{n} P(X_j|Pa(X_j), C_i) & si \quad X_j \ admet \ des \ parents. \\ P(C_i)\prod_{j=1}^{n} P(X_j|C_i) & \text{sinon.} \end{cases} \quad \textbf{(IV.98)}$$

Avec

- $P(X_j|Pa(X_j), C_i)$ est la fréquence d'apparition de X_j en connaissant ses parents $Pa(X_j)$dans la classe C_i si c'est le cas de variables discrètes données par les équations (IV.91) , (IV.93), (IV.94) ou (IV.95).

Ou bien :

- $P(X_j|Pa(X_j), C_i)$ est une distribution gaussienne donnée par l'équation (IV.96).

Dans le cas ou les probabilités conditionnelles ne sont pas exactes, effectuer une inférence exacte avec ces valeurs approximatives n'est plus probant. Il est alors intéressant d'effectuer une inférence approchée, en utilisant d'autres algorithmes tels que l'algorithme de Métropolies Hastings, l'échantillonneur de Gibbs, Loopy belief propagation [***221***].

IV.5.4.7 Structures des réseaux bayésiens pour la classification

Dans les tâches de classification, une variable précise correspond à la classe qu'il faut reconnaître à partir des autres variables (les caractéristiques). Dans ce cas, le nœud parent est considéré comme une variable non observée précisant à quelle classe appartient chaque objet alors que les nœuds enfants sont des variables observées correspondant aux différents attributs caractérisant cet objet. Plusieurs méthodes d'apprentissage vont donc proposer des structures où ce nœud classe aura un rôle central [***222***], [***223***], [***224***].

Plusieurs modèles sont conçus dans ce but. Parmi ces réseaux, on peut citer le réseau Bayésien naïf qui est le plus simple, le réseau bayésien augmenté par n'importe qu'elle structure ou par une structure arborescente et autres. Les réseaux Bayésiens ont une structure simple et unique qui comprend deux niveaux. Le premier niveau contient un seul nœud parent et le second comporte plusieurs enfants avec la forte hypothèse naïve d'indépendance conditionnelle des enfants conditionnellement au parent. Ils sont largement utilisés pour résoudre des problèmes de classification [***225***].

IV.5.4.7.1) Structure de Bayes naïve

Le classificateur de Bayes naïf correspond à la structure la plus simple en posant l'hypothèse que les nœuds caractéristiques sont indépendantes conditionnellement du nœud de la classe. Cela mène à la structure type de la Figure IV-24 suivante. Cette structure, pourtant très simple, donne de très bons résultats dans de nombreuses applications [***226***].

Ci désigne le nœud classe et i la i ième classe. X_j est est le j ième attribut ou paramètre du nœud j.

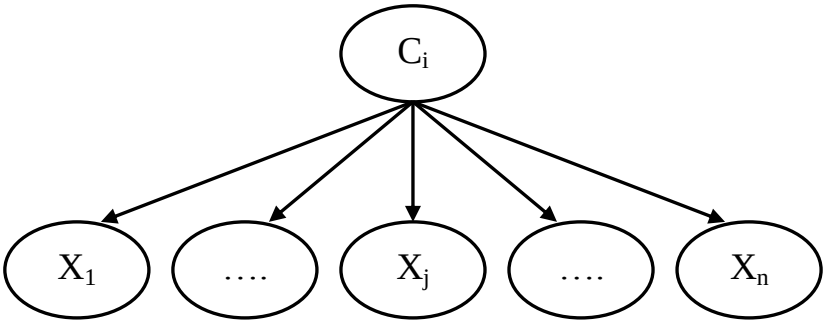


Figure IV-24: Exemple de structure d'un réseau bayésien naïf.

Ce classificateur est connu pour ses performances malgré sa simplicité et il est préférable à des techniques beaucoup plus sophistiquées même lorsque l'hypothèse

d'indépendance est violée. L'hypothèse d'indépendance des variables permet d'écrire la probabilité a posteriori de chaque classe comme l'indique l'équation suivante :

$$P(C_i|X) = P(C_i)\prod_{j=1}^{n} P(X_j|C_i) \quad \textbf{(IV.99)}$$

Par conséquent, en présence d'un ensemble d'apprentissages, la seule opération à faire est de calculer les probabilités conditionnelles en appliquant la règle de décision d comme suit:

$$\begin{aligned} d(X) &= \arg\max_{C_i} P(C_i|X) \\ &= \arg\max_{C_i} P(X|C_i)\,P(C_i) \\ &= \arg\max_{C_i} P(C_i)\prod_{j=1}^{n} P(X_j|C_i) \end{aligned} \quad \textbf{(IV.100)}$$

L'hypothèse d'indépendance entre les attributs utilisés dans le réseau bayésien naïf n'est pas toujours vraie (hypothèse naïve). Il existe différentes techniques pour assouplir cette hypothèse [***227***]. Elles consistent à identifier les dépendances conditionnelles entre les attributs. Ce qui amène à la nouvelle structure augmenté de la structure de bayes naïve.

IV.5.4.7.2) Structure augmentée

Afin d'alléger l'hypothèse d'indépendance conditionnelle des caractéristiques, il a été suggéré d'augmenter la structure de bayes naïve en ajoutant des liens entre certaines caractéristiques ou attributs [***228***], [***229***], [***230***].

Parmi les différentes méthodes avancées pour augmenter le réseau bayésien naïf, citons TANB (Tree Augmented Naive Bayes) qui utilise une structure naïve entre la classe et les caractéristiques et un arbre reliant toutes les caractéristiques. Cette structure augmentée est illustrée par la Figure IV-25 suivante :

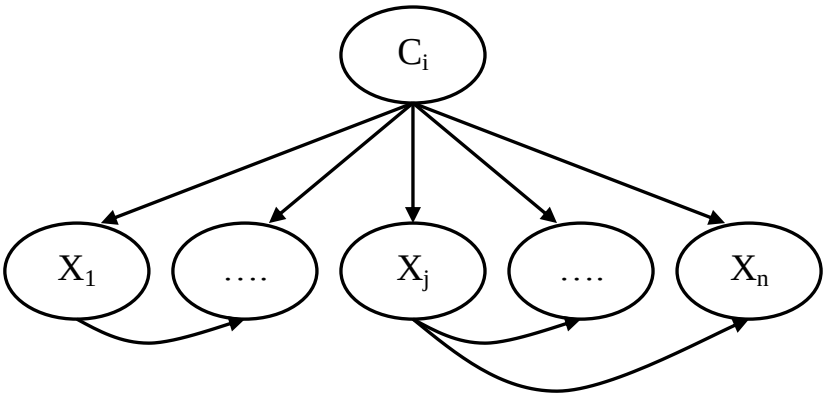


Figure IV-25: Exemple de structure d'un réseau bayésien naïf augmenté par un arbre.

D'autres types de réseaux bayésiens ayant des structures plus complexes peuvent être rencontrés et utilisés pour la classification d'images. Le réseau bayésien de Bayes naïf est utilisé dans cette thèse en raison de sa simplicité ainsi que sa robustesse.

IV.6 Conclusion

Dans ce chapitre, nous avons discuté et développé la structure formelle du système d'annotation automatique d'images, et nous avons présenté une étude détaillée des différentes techniques et méthodes utilisées. En effet, deux méthodes de segmentation ont été présentées en détaillant leur algorithme. Ensuite, plusieurs méthodes d'extraction d'attributs caractérisant des objets contenus dans une image ont été aussi détaillées. Afin de pouvoir finalement annoter ces objets, ils doivent être bien classifiés pour pouvoir choisir les mots clé convenables. Ainsi, plusieurs types de classificateurs ont été étudiés afin d'en choisir le plus convenable ou de les combiner pour peu de complémentarité. La réalisation et les résultats d'un tel système feront l'objet du chapitre suivant.

Chapitre V REALISATION DU SYSTEME D'ANNOTATION AUTOMATIQUE D'IMAGES

« L'image est un modèle de la réalité. »

[Ludwig Wittgenstein]

Contenu du Chapitre

V.1 Introduction

Dans ce chapitre, nous décrivons la démarche expérimentale adoptée et les expériences réalisées pour mettre en œuvre et évaluer le système d'annotation automatique d'images conçu et présenté dans le chapitre précédent. Les résultats obtenus pour chaque expérience seront présentés et discutés afin de tirer des conclusions sur l'efficacité du système d'annotation automatique d'images réalisé, ainsi que les difficultés et les défis qui doivent être relevés.

V.2 Mise en œuvre

V.2.1 Démarche expérimentale

Le système proposé consiste à trouver automatiquement, pour chaque image, les termes d'annotations qui décrivent son contenu visuel. La démarche expérimentale adoptée est représentée par le schéma fonctionnel donné par la Figure V-1.

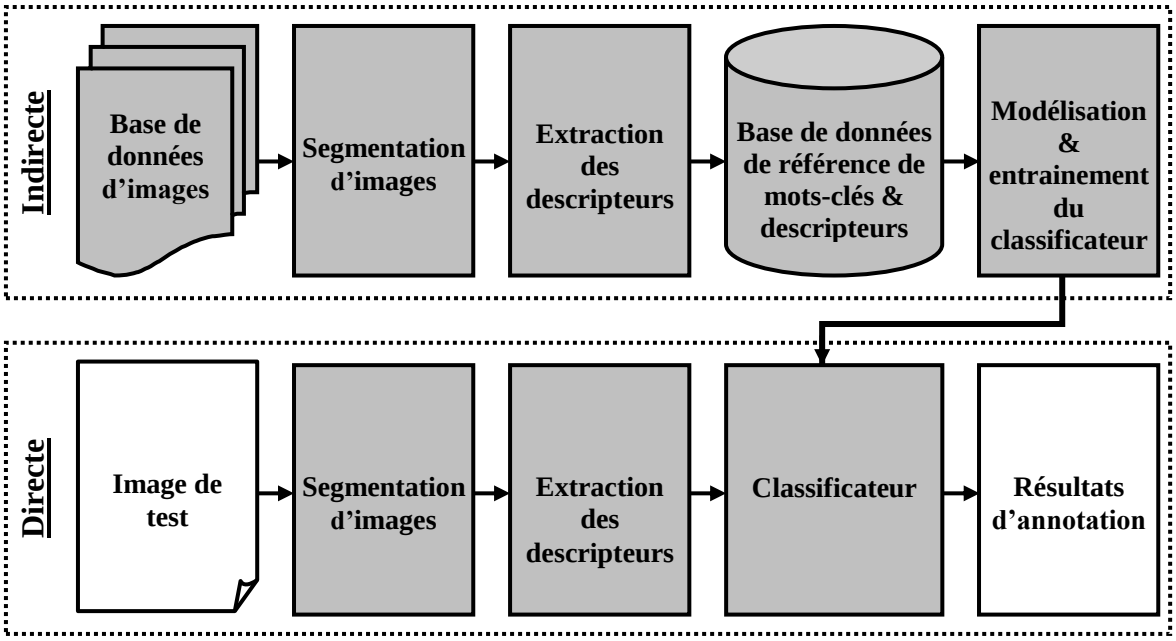


Figure V-1 : Schéma fonctionnel du système d'annotation automatique d'images proposé.

Ce système comprend deux parties essentielles : la partie d'annotation indirecte (annotation manuelle) et la partie d'annotation directe (annotation automatique). Il comporte plusieurs phases qui sont : la segmentation en région, l'extraction des descripteurs et le choix des mots clés convenables à l'aide d'un classificateur déjà entrainé pour ce but. Les différentes techniques utilisées dans ces phases sont déjà présentées en détail dans le chapitre précédent.

V.2.1.1 Partie d'annotation indirecte

La première partie indirecte permet l'entrainement et la modélisation des classificateurs afin d'avoir une correspondance entre les mots-clés d'images et les vecteurs descripteurs qui les représentent. Elle se compose de plusieurs modules, à savoir :

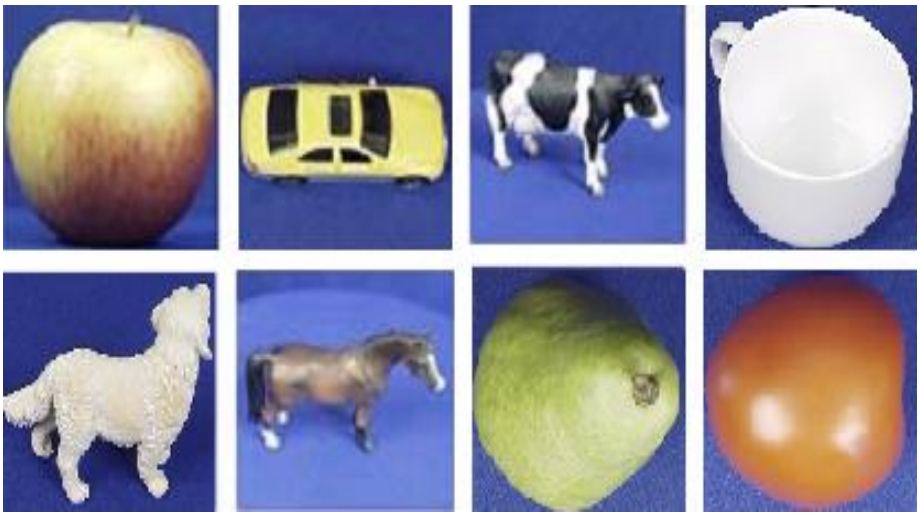
- Base de données d'images ;

- Segmentation d'images ;
- Extraction des descripteurs ;
- Modélisation et entrainement des classificateurs.

V.2.1.1.1) Base de données d'images

Notre objectif est d'étudier comment automatiser l'annotation d'images en utilisant un vocabulaire composé de mots-clés afin de réduire le fossé sémantique existant entre le contenu visuel de bas niveau et les concepts sémantiques de l'image. Le choix d'une base d'images adaptée à ce genre de problématique est alors important et constitue l'étape primaire de l'annotation d'images. Il existe beaucoup de bases d'images utilisées pour la reconnaissance d'objets et l'annotation d'images. Parmi les plus couramment utilisées, citons la base de données ETH-80 [***231***], la base de données COIL-100 [***232***], la base de données PASCAL VOC [***233***], la base de données CALTECH-101 [***234***], la base de données Caltech-256 [***235***], ou encore la base de données LabelMe [***20***].

Dans notre travail, les expériences d'annotation d'images sont réalisées en utilisant la base de données ETH-80 contenant un ensemble d'images en couleur de 8 objets différents [***231***], la base de données COIL-100 qui contient des images en couleur de 100 objets différents [***232***] et la base de données NATURE qui contient des images en couleur de 6 éléments de la nature. La Figure V-2 montre quelques exemples d'images de ces bases.



a)

b)

c)

Figure V-2 : Exemples d'objets de la base de données d'images:
a) ETH-80, b) COIL-100, c) NATURE.

La base de données des caractères amazighs [***236***] est aussi utilisée dans un premier temps pour d'une part, se familiariser avec les différents descripteurs et classificateurs intégrés dans notre système et pour d'autre part, tester leur efficacité en reconnaissance de formes [***117***], [***237***], [***238***]. De bons résultats ont été obtenus (voir **Annexe A**).

V.2.1.1.2) Segmentation d'images

La première phase du système d'annotation automatique d'images proposé consiste à diviser l'image d'entrée en un ensemble d'objets séparés. Pour ce faire, la segmentation par croissance de région ainsi que la segmentation par la méthode des k-moyennes ont été choisies et utilisées. Les deux algorithmes de segmentation ont été présentés et étudiés dans le chapitre précédent.

Un exemple de segmentation d'images, en utilisant l'algorithme de segmentation par croissance de régions, est présenté sur la Figure V-3 pour certains objets de la base de données d'images ETH-80 et la base de données d'images COIL-100.

a) b)

Figure V-3 : Exemple de segmentation d'images par croissance de région résultant de la base de données d'images: a) ETH-80, b) COIL-100.

Un autre exemple de segmentation d'images, en utilisant l'algorithme des K-moyennes, est présenté sur la Figure V-4.

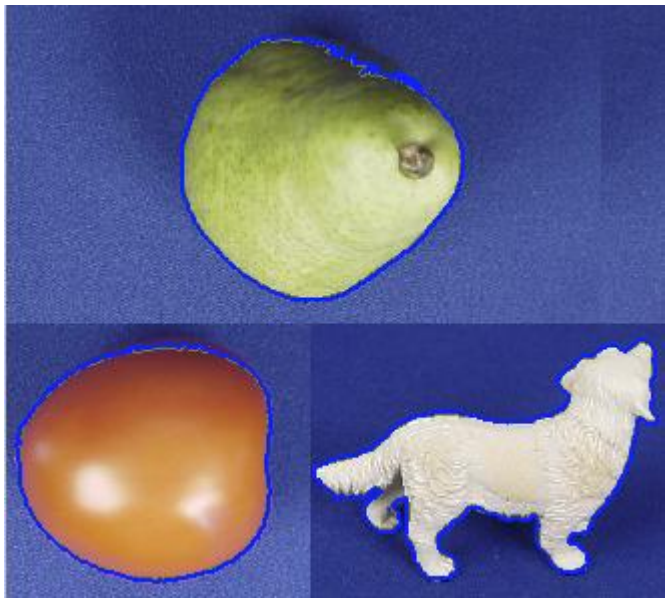


Figure V-4 : Exemple de segmentation d'images par l'algorithme des K-moyennes.

Il se trouve que lors de la segmentation, un objet se décompose en plusieurs régions, dû au prédicat d'homogénéité utilisé pour contrôler la segmentation d'images (voir paragraphe IV.3.1, Chapitre VI, page : 38), ce qui induit des erreurs au niveau de la classification. Pour résoudre ce problème, nous avons développé une méthode qui consiste à regrouper les régions adjacentes. Les résultats de cette proposition sont présentés et analysés dans la dernière expérience du système d'annotation automatique d'images (voir paragraphe V.2.2.4, Chapitre V, page : 155).

V.2.1.1.3) Extraction d'attributs d'images

Après avoir divisé l'image d'entrée en un ensemble de régions représentant les objets contenus dans l'image, la phase suivante consiste à extraire un ensemble de caractéristiques réduites capable de représenter les objets de l'image d'une manière efficace, indépendante et résistante aux transformations géométriques telles que la rotation, la translation et le changement d'échelle. Dans nos expériences, pour chaque région qui représente un objet d'un plan de couleur de l'image requête, les caractéristiques extraites sont :

- 7 éléments pour les moments invariants de Hu (Hu1, Hu2, Hu3, Hu4, Hu5, Hu6, Hu7).

- 9 éléments en utilisant l'ordre 4 pour les moments de Zernike (Z00, Z11, Z20, Z22, Z31, Z33, Z40, Z42, Z44).

- 10 éléments en utilisant l'ordre 3 pour les moments de Legendre (L00, L01, L02, L03, L10, L11, L12, L20, L21, L30).

- 16 éléments par canal d'images pour les histogrammes de couleur d'images.

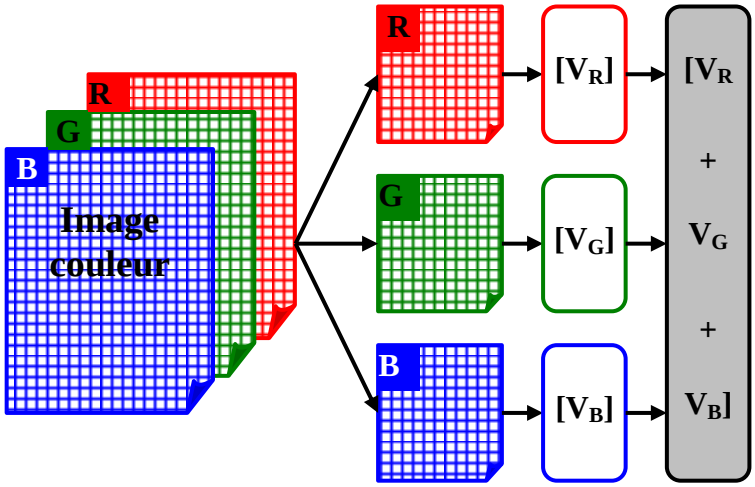


Figure V-5 : Processus de calcul des descripteurs d'une image couleur pour les trois canaux de couleur R, G et B.

Pour toutes les expériences, ces descripteurs sont calculés pour chaque canal de couleurs d'images et ils sont regroupés et fusionnés en un seul vecteur de façon à ce qu'ils représentent une entrée unique aux classificateurs. La Figure V-5 représente le processus de calcul de ces descripteurs pour les trois canaux de couleur R, G et B d'une image couleur. Ainsi, le vecteur descripteur résultant est de 21 éléments pour les moments de Hu, 27 éléments pour les moments de Zernike, 30 éléments pour les moments de Legendre et 48 éléments pour les histogrammes de couleur dans le cas des 3 canaux de couleurs de l'image.

En plus de ces descripteurs, nous avons utilisé le descripteur global GIST et le descripteur de texture.

Le descripteur global GIST est construit en combinant les amplitudes obtenues à la sortie de K filtres de Gabor à différentes échelles E et orientations O. Chaque image est subdivisée en N * N blocs (N compris entre 2 et 16). Ceci donne un vecteur de dimension N * N * K * E * O = 128 (dans notre cas, N=4, K=4, E=4, O=2). Pour réduire la taille du vecteur descripteur, l'analyse en composante principale peut être appliquée. La Figure V-6 donne un exemple de descripteur GIST d'une image requête.

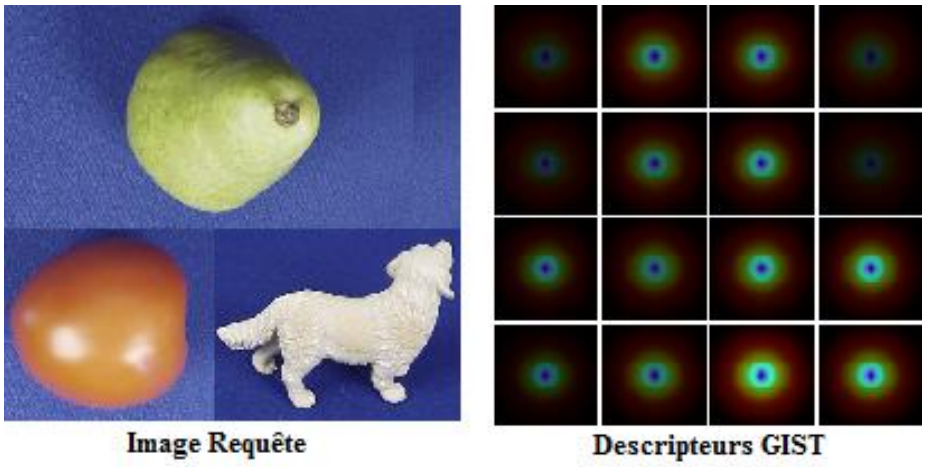


Figure V-6 : Exemple de descripteur GIST d'une image requête

Pour le descripteur de texture, nous avons calculé un vecteur contenant 56 composantes des caractéristiques résultant des 14 indices d'Haralick pour chacune des 4 directions de proximité d'une image en niveaux de gris (matrices de cooccurrence, voir paragraphe IV.4.3.2.5)b), Chapitre VI, page : 69). Lorsqu'on utilise une image en couleur, le vecteur se compose de 168 composantes.

Tous les descripteurs extraits sont regroupés avec les mots-clés correspondants dans une base de données de référence qui va servir ensuite pour l'entrainement et la modélisation des classificateurs (Réseaux de neurones, réseaux bayésiens et les SVM multi-classes) lors de l'apprentissage et pour calculer la similarité entre l'image requête et les mots-clés lors de la classification par les k-plus proches voisins.

V.2.1.1.4) Apprentissage et entrainement des classificateurs

La base de données de référence, qui contient les mots-clés et les vecteurs descripteurs correspondants, servira pour l'apprentissage supervisé et l'entrainement des réseaux de neurones, des réseaux bayésiens et des SVM multi classes afin de rechercher et générer un modèle reliant et faisant la correspondance entre les descripteurs et les mots-clés convenables. L'apprentissage dans le cas des réseaux de neurones et des SVM multi classes, appelés classificateurs discriminatifs, a pour objectif de tracer des frontières précises entre des ensembles de points représentant les catégories. Dans le cas des réseaux bayésiens, appelés classificateurs génératifs, l'apprentissage consiste à étudier d'abord la distribution des caractéristiques et la probabilité à priori pour chacune des classes à classifier pour pouvoir inférer la probabilité à posteriori qui est utilisée dans la décision.

Les modèles générés après la phase d'apprentissage sont utilisés ultérieurement lors de la phase d'annotation d'une image requête.

V.2.1.2 Partie d'annotation directe

Cette partie consiste, en utilisant les classificateurs entrainés, à attribuer les mots-clés les plus probables à une image requête.

Pour atteindre cet objectif, l'image requête est segmentée en régions qui représentent les objets contenu dans l'image, ensuite les caractéristiques de chaque région sont calculées et extraites de l'image. Ces caractéristiques sont fournies finalement à l'entrée des classificateurs (les réseaux de neurones, les réseaux bayésiens, les SVM multi-classes et les k plus proches voisins) qui permettent de décider et de choisir les mots-clés appropriés pour l'annotation.

Afin d'élaborer un système d'annotation automatique d'images performant, nous avons réalisé plusieurs expériences d'annotation d'images selon cette démarche expérimentale que nous venons de présenter. Ces expériences sont présentées ci-dessous.

V.2.2 Expériences

Après avoir implémenté le système proposé, plusieurs expériences ont été faites en utilisant les trois bases de données d'images ETH-80, COIL-100 et NATURE. Ces expériences sont réalisées sur un ordinateur personnel Core 2 Duo (CPU cadencé à 1.80 GHz et 2 Go de mémoire RAM) en utilisant Matlab R2012b. Pour les trois bases de données d'images, nous avons utilisé 40 images pour l'apprentissage et 40 autres images pour les tests. Le taux d'annotation, déjà présenté précédemment par l'Equation (III.1), est calculé à partir du nombre d'images annotées correctement sur le nombre d'images utilisées pour le test.

La première expérience consiste à utiliser individuellement les descripteurs et les classificateurs, la deuxième expérience consiste à combiner ou fusionner les descripteurs, la troisième expérience consiste à combiner les classificateurs et la quatrième expérience consiste à tester l'effet de regroupement des régions adjacentes sur l'efficacité de l'annotation automatique d'images.

Les résultats de ces expériences sont présentés et analysés dans cette section. Il faut noter aussi que ces descripteurs et classificateurs ont été appliqués dans un premier temps

pour la reconnaissance des caractères Tifinagh avant d'être appliqués pour l'annotation d'images couleurs [***117***], [***237***], [***238***], [***239***], [***240***] (voir **Annexe A**).

V.2.2.1 Expérience 1 : Utilisation individuelle des classificateurs et descripteurs

Dans cette première approche expérimentale, les classificateurs et les descripteurs sont utilisés individuellement avec chaque approche de segmentation automatique d'images. Le principe utilisé est représenté par le schéma bloc illustré sur la Figure V-7.

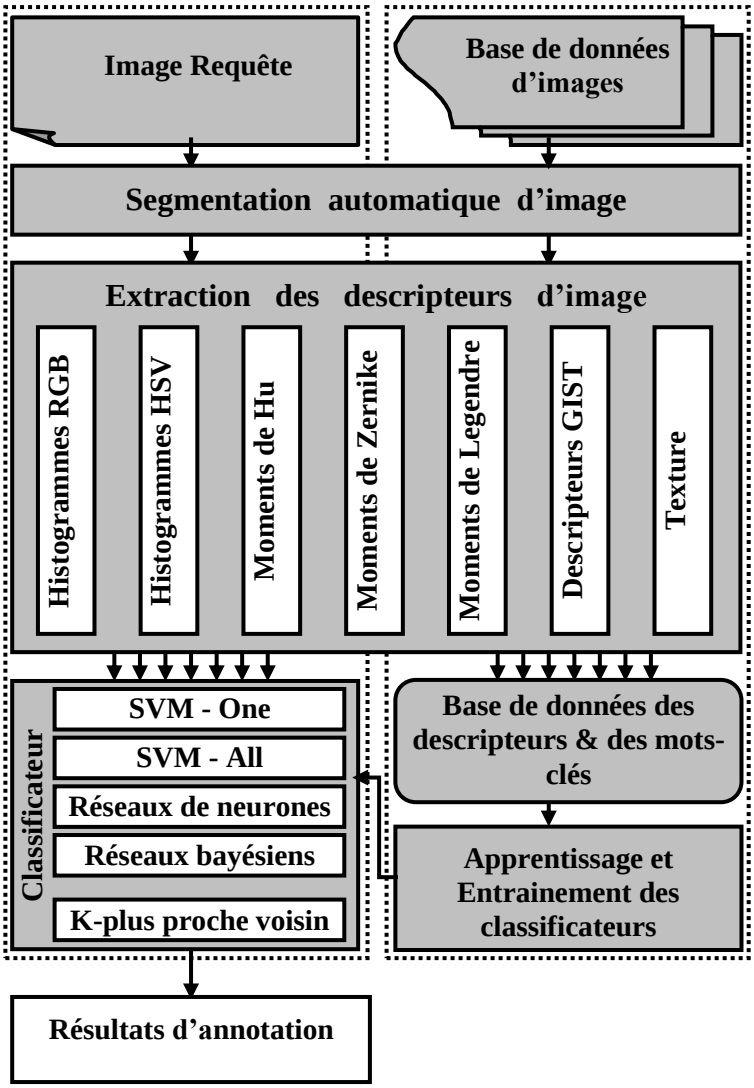


Figure V-7 : Principe du système d'annotation automatique d'image basé sur l'utilisation individuelle des descripteurs et classificateurs.

Les résultats de cette expérience nous permettront d'avoir une idée générale sur l'efficacité de chaque descripteur et chaque classificateur afin d'en choisir les meilleurs et les plus convenables pour les expériences qui suivent.

V.2.2.1.1) Résultats

Les résultats d'annotation pour chaque classificateur et chaque descripteur, en utilisant les bases de données d'images ETH-80, COIL-100 et NATURE, sont présentés dans les Tableaux V-1 et V-2.

Tableau V-1 : Taux d'annotation de chaque classificateur et chaque descripteur en utilisant les bases de données d'images ETH-80 et COIL-100.

Algorithme de Segmentation	Base de données	Descripteur	Classificateur				
			Discriminatif				Génératif
			K-NN	SVM-One	SVM-All	Neural Network	Bayesian Network
Croissance de Région (Region Growing)	ETH-80	Hu	59.80%	53.75%	58.13%	61.53%	70.00%
		Zernike	68.41%	62.50%	65.00%	69.70%	76.32%
		Legendre	78.55%	73.75%	75.00%	79.91%	82.50%
		RGB	56.25%	57.92%	55.00%	65.00%	71.67%
		HSV	50.00%	46.25%	41.25%	55.00%	66.25%
		Texture	47.50%	43.50%	42.50%	53.50%	58.00%
		GIST	58.75%	55.00%	52.50%	60.00%	65.50%
	COIL-100	Hu	50.00%	43.75%	43.75%	61.25%	65.00%
		Zernike	71.25%	60.00%	65.00%	73.75%	75.50%
		Legendre	73.50%	70.00%	75.00%	77.50%	80.00%
		RGB	63.75%	60.00%	58.75%	65.00%	67.50%
		HSV	61.25%	60.00%	58.75%	63.75%	72.50%
		Texture	45.00%	42.50%	40.50%	50.00%	55.50%
		GIST	55.00%	52.50%	50.00%	58.50%	65.00%
K-Moyennes (K-Means)	ETH-80	Hu	56.50%	52.00%	54.75%	58.00%	66.00%
		Zernike	64.50%	59.00%	61.00%	65.75%	72.00%
		Legendre	74.00%	69.50%	70.75%	75.00%	78.00%
		RGB	53.00%	54.50%	52.00%	61.00%	67.50%
		HSV	47.25%	43.50%	39.00%	51.75%	62.50%
		Texture	44.75%	40.50%	40.00%	50.50%	55.00%
		GIST	55.50%	51.75%	49.50%	56.50%	62.50%
	COIL-100	Hu	47.00%	41.25%	41.25%	57.75%	61.25%
		Zernike	67.25%	56.50%	61.25%	69.50%	71.25%
		Legendre	69.25%	66.00%	70.75%	73.00%	75.50%
		RGB	60.00%	56.50%	55.50%	61.25%	63.50%
		HSV	57.75%	56.50%	55.50%	60.00%	68.25%
		Texture	42.50%	40.00%	38.00%	47.25%	52.50%
		GIST	52.00%	49.50%	47.25%	55.25%	61.25%

Les résultats d'annotation pour chaque classificateur et chaque descripteur, en utilisant la base de données d'images NATURE, sont aussi présentés dans le Tableau V-2.

Tableau V-2 : Taux d'annotation de chaque classificateur et chaque descripteur pour des images de la base de données d'images NATURE.

Descripteur	Classificateur			
	Discriminatif			Génératif
	SVM-One	SVM-All	Neural Network	Bayesian Network
Legendre	66.67%	66.67%	66.67%	73.33%
RGB	66.67%	66.67%	60.00%	66.67%
Texture	66.67%	66.67%	70.00%	80.00%

Les résultats d'annotation pour chaque classificateur et chaque descripteur, en utilisant les bases de données d'images ETH-80 et COIL-100, sont aussi illustrés sur la Figure V-8.

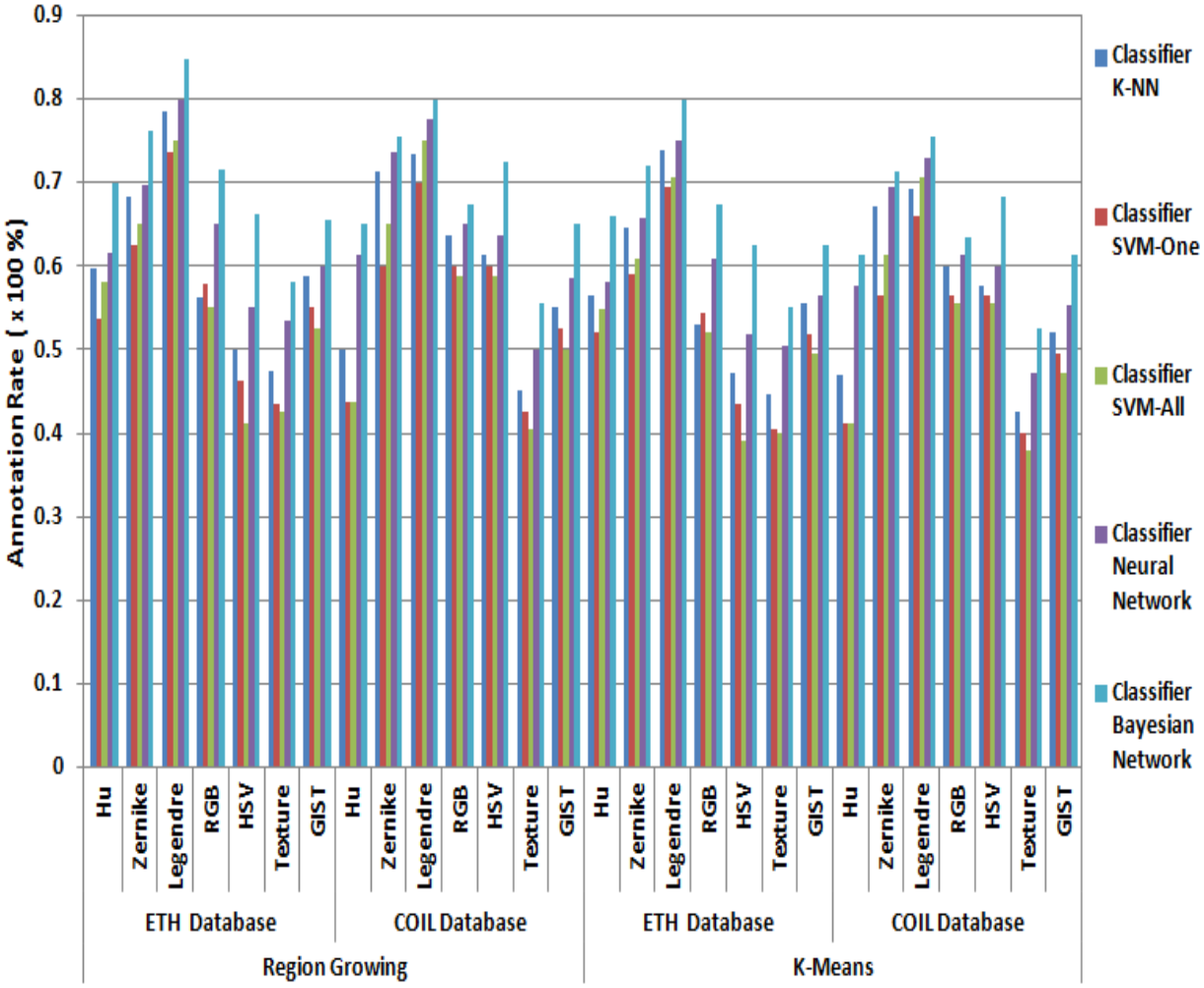


Figure V-8 : Taux d'annotation dans le cas d'usage individuel des classificateurs et descripteurs pour les bases de données d'images ETH-80 et COIL-100.

Les résultats d'annotation pour chaque classificateur et chaque descripteur, en utilisant la base de données d'images NATURE, sont aussi illustrés sur la Figure V-9.

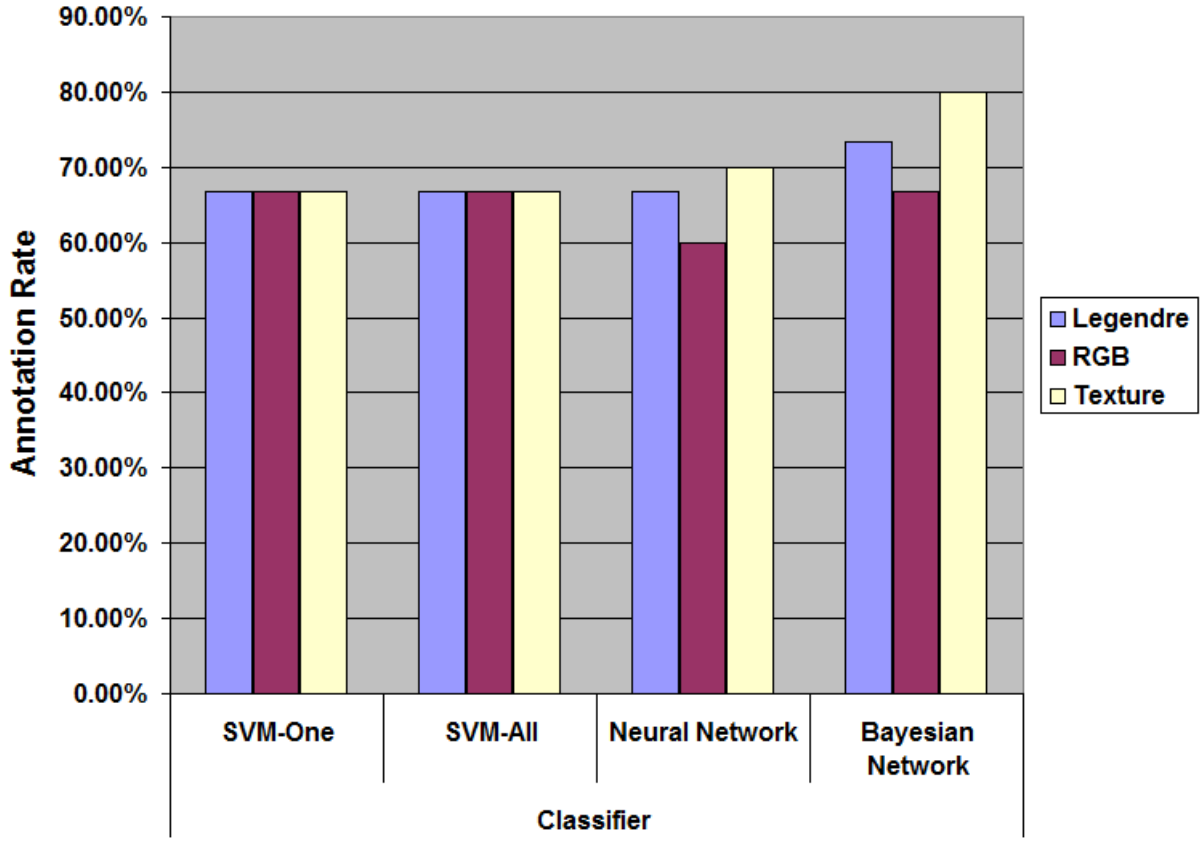


Figure V-9 : Taux d'annotation dans le cas d'usage individuel des classificateurs et descripteurs pour la base de données d'images NATURE.

La matrice de confusion donnée par la Figure V-10 montre les objets mal annotés (indiqués par la couleur rouge) dans le cas d'utilisation des moments de Legendre comme descripteur et les réseaux bayésiens comme classificateur pour des images de la base de données ETH-80.

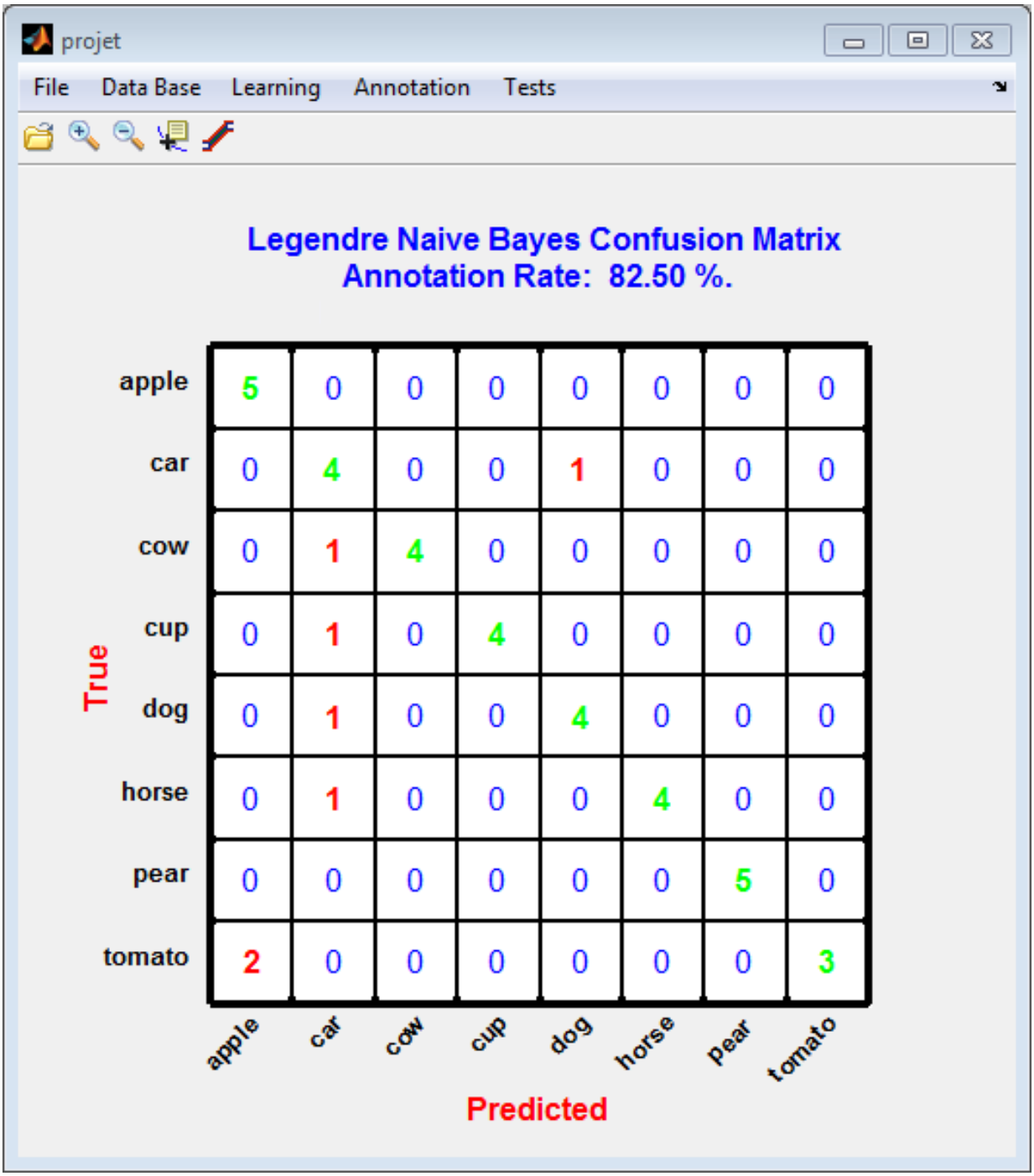


Figure V-10 : Matrice de confusion dans le cas d'utilisation des réseaux bayésiens et les moments de Legendre pour des images de la base de données ETH-80.

La Figure V-11 montre la matrice de confusion dans le cas d'utilisation des moments de Legendre comme descripteur et le réseau bayésien comme classificateur pour des images à partir de la base de données COIL-100.

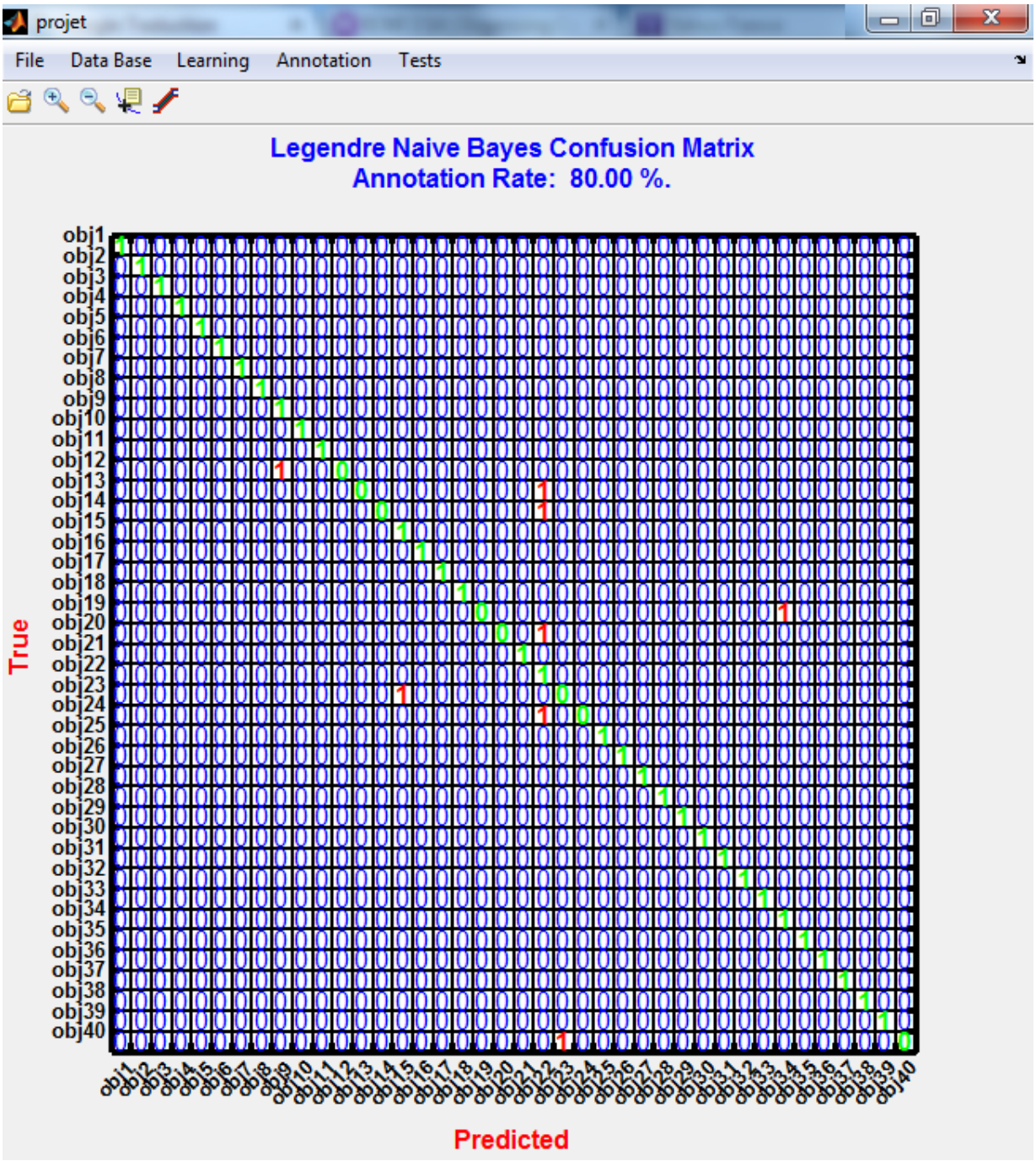


Figure V-11 : Matrice de confusion dans le cas d'utilisation des réseaux bayésiens et les moments de Legendre pour des images de la base de données COIL-100.

La Figure V-12 montre la matrice de confusion dans le cas d'utilisation de la texture (indices d'Haralik basés sur les matrices de coocurrences chromatiques) comme descripteur et les réseaux bayésiens comme classificateur pour des images à partir de la base de données d'images NATURE.

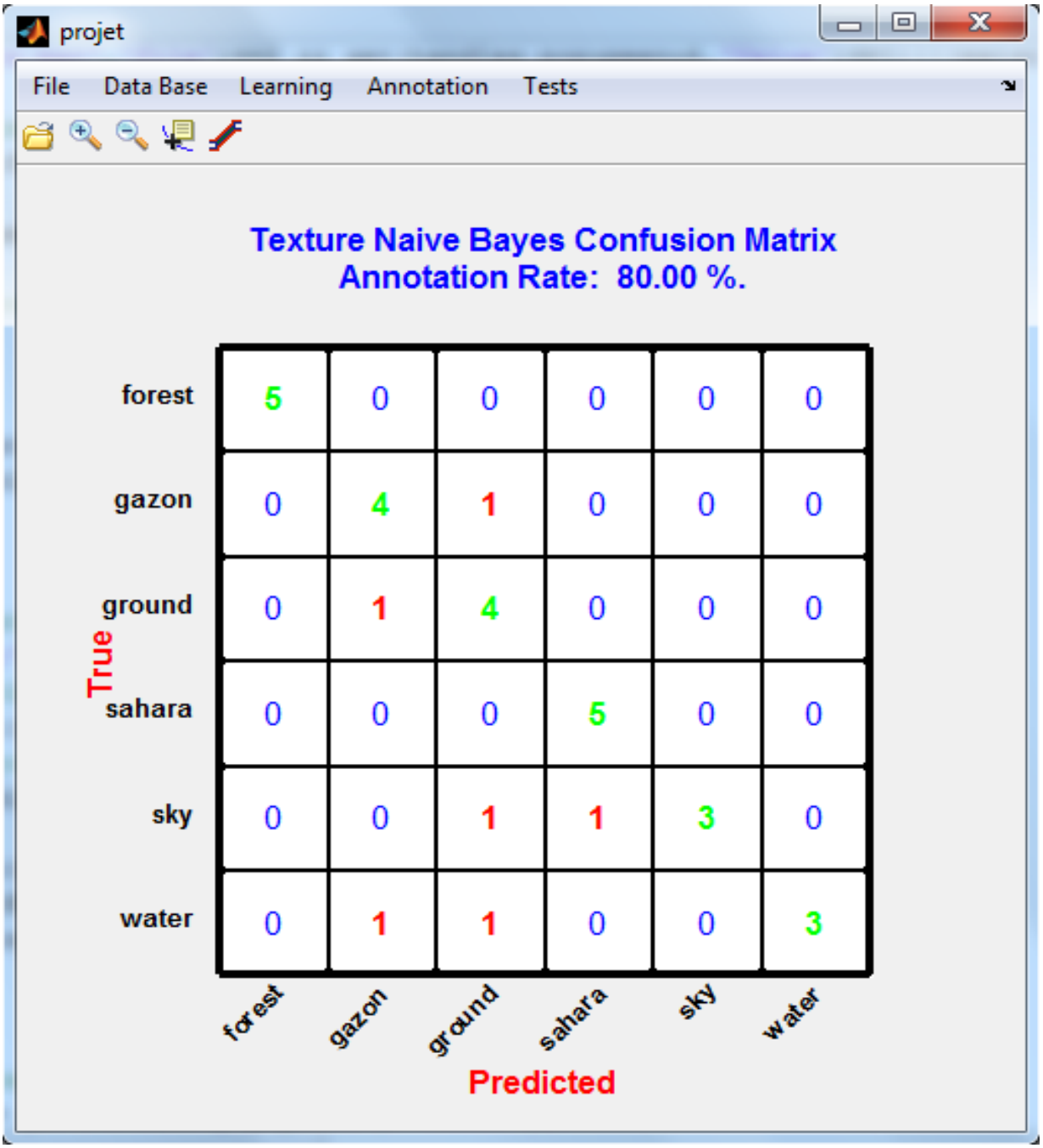


Figure V-12 : Matrice de confusion dans le cas d'utilisation de la texture et le réseau bayésien pour des images à partir de la base de données d'images NATURE.

La Figure V-13 montre la matrice de confusion dans le cas d'utilisation des moments de Legendre comme descripteur de forme et le réseau bayésien comme classificateur génératif pour des images à partir de la base de données d'images NATURE.

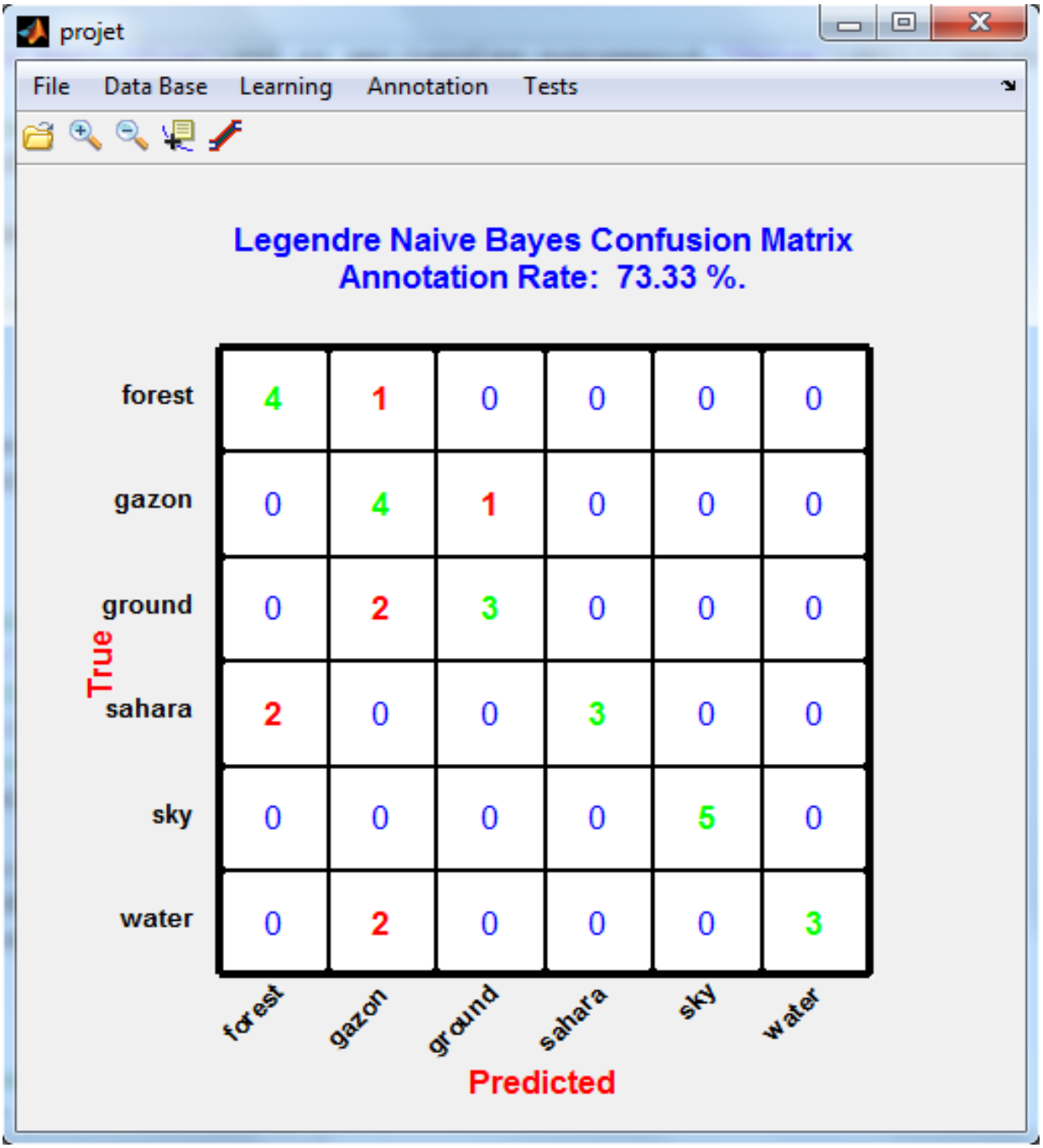


Figure V-13 : Matrice de confusion dans le cas d'utilisation des moments de Legendre et le réseau bayésien pour des images à partir de la base de données d'images NATURE.

Pour avoir également une idée globale sur les performances et le comportement de chaque descripteur et chaque classificateur pour chaque mot-clé de la base de données considérée, les résultats d'annotations pour chaque mot-clé de la base de données d'images ETH-80 sont donnés dans le Tableau V-3 suivant :

Tableau V-3 : Résultats d'annotations pour chaque mot-clé de la base de données d'images ETH-80.

Classificateur	Object	Descripteur						
		Hu	Zernike	Legendre	RGB	HSV	Texture	GIST
K-NN	Apple	69.12%	76.84%	90.93%	64.50%	52.00%	52.00%	68.50%
	Car	67.03%	72.87%	81.91%	50.50%	42.50%	36.50%	52.50%
	Cow	39.82%	49.13%	60.10%	38.50%	38.00%	39.00%	41.50%
	Cup	62.07%	70.37%	80.02%	61.00%	56.00%	50.00%	63.50%
	Dog	60.87%	70.93%	79.10%	61.50%	54.00%	50.50%	63.00%
	Horse	46.80%	55.10%	63.80%	40.50%	37.50%	38.50%	43.00%
	Pears	66.80%	76.94%	87.85%	67.00%	60.50%	60.00%	68.50%
	Tomato	65.90%	75.10%	84.70%	66.50%	59.50%	53.50%	69.50%
SVM-One	Apple	62.50%	70.00%	85.25%	66.50%	48.00%	47.75%	64.00%
	Car	53.00%	66.25%	76.00%	52.00%	39.25%	33.50%	49.25%
	Cow	38.50%	46.00%	56.50%	39.75%	35.50%	35.75%	39.00%
	Cup	55.50%	64.00%	75.00%	62.75%	51.50%	45.50%	59.50%
	Dog	54.50%	65.00%	74.25%	63.25%	50.00%	46.25%	59.00%
	Horse	42.50%	50.25%	59.00%	41.75%	34.75%	35.25%	40.25%
	Pears	62.00%	70.25%	83.50%	69.00%	56.00%	55.00%	64.00%
	Tomato	61.50%	68.25%	80.50%	68.50%	55.00%	49.00%	65.00%
SVM-All	Apple	67.50%	72.75%	86.50%	63.00%	43.00%	46.75%	61.00%
	Car	57.25%	69.00%	77.25%	49.50%	35.00%	32.75%	47.00%
	Cow	41.75%	48.00%	57.50%	37.75%	31.75%	35.00%	37.25%
	Cup	60.00%	66.50%	76.25%	59.50%	45.25%	44.50%	56.75%
	Dog	59.00%	67.50%	75.50%	60.00%	44.50%	45.00%	56.50%
	Horse	46.00%	52.25%	60.00%	39.75%	31.50%	34.50%	38.50%
	Pears	67.00%	73.00%	85.00%	65.50%	50.00%	53.75%	61.00%
	Tomato	66.50%	71.00%	82.00%	65.00%	49.00%	47.75%	62.00%
Neural Network	Apple	71.20%	78.40%	92.13%	74.50%	57.25%	59.00%	69.75%
	Car	68.53%	74.37%	83.11%	58.50%	46.75%	41.25%	53.75%
	Cow	41.32%	50.00%	61.37%	44.75%	42.25%	44.00%	42.75%
	Cup	63.27%	71.20%	81.20%	70.25%	60.25%	56.00%	64.75%
	Dog	62.57%	72.30%	80.41%	71.00%	59.25%	56.50%	64.50%
	Horse	48.60%	56.40%	65.28%	47.00%	42.50%	43.50%	44.00%
	Pears	69.29%	78.40%	89.38%	77.50%	66.50%	67.75%	69.75%
	Tomato	67.49%	76.52%	86.40%	76.50%	65.25%	60.00%	70.75%
Bayesian Network	Apple	81.00%	85.75%	92.50%	82.00%	69.00%	90.00%	100.00%
	Car	78.00%	81.50%	85.00%	64.50%	56.25%	22.00%	22.00%
	Cow	47.00%	54.75%	62.50%	49.25%	51.25%	20.00%	22.00%
	Cup	72.00%	78.00%	82.50%	77.50%	72.50%	70.00%	80.00%
	Dog	71.25%	79.25%	82.50%	78.25%	71.25%	60.00%	90.00%
	Horse	55.25%	61.75%	70.00%	51.75%	51.25%	40.00%	30.00%
	Pears	79.00%	86.00%	92.50%	85.50%	80.00%	80.00%	100.00%
	Tomato	76.75%	83.75%	92.50%	84.50%	78.50%	82.00%	80.00%

V.2.2.1.2) Analyse des résultats et conclusion

L'analyse des Tableaux V-1, V-2 et V-3 et des Figures V-8, V-9, V-10, V-11, V-12 et V-13 montre bien que le taux d'annotation automatique d'images dépend du type de classificateur, de la nature du descripteur, de la méthode de segmentation d'image et du nombre de mots-clés utilisé.

En effet, nous avons constaté que :

- Pour chaque descripteur, le meilleur taux d'annotation est obtenu par le classificateur génératif qui est le réseau bayésien.

- Le meilleur taux d'annotation obtenu par les classificateurs discriminatifs est celui qui est réalisé par les réseaux de neurones. Ce taux s'approche bien de celui obtenu par les réseaux bayésiens, en particulier dans le cas de la base de données d'images COIL-100 contenant plus d'objets que la base de données d'images ETH-80.

- Pour les deux bases de données d'images ETH-80 et COIL-100, contenant des objets ayant en général des formes bien définies, les descripteurs de Legendre permettent d'obtenir, pour chaque classificateur, le meilleur taux d'annotation.

- Pour la base de données d'images NATURE, le meilleur taux d'annotation est obtenu, pour chaque classificateur, par le descripteur de texture (indices d'Haralik basés sur les matrices de cooccurrences chromatiques).

- Certains descripteurs réalisent avec les réseaux de neurones des taux d'annotation meilleurs que ceux obtenus avec les réseaux bayésiens ; et l'inverse pour d'autres descripteurs (voir Tableau V-3).

- Certains objets sont confondus avec d'autres objets (voir les matrices de confusion sur les Figures V-10, V-11, V-12 et V-13).

- Le meilleur taux d'annotation, pour chaque descripteur et chaque classificateur, est obtenu en utilisant la méthode de segmentation par croissance de région.

- Les taux d'annotation de certains descripteurs augmentent avec l'augmentation du nombre de mots clés de la base de données d'images et ceux d'autres descripteurs diminuent.

En se basant sur l'ensemble des constatations et remarques citées ci-dessus, nous pouvons conclure que pour mieux annoter les images d'une large base de données d'images contenant des objets qui ne peuvent être distingués que soit par leurs formes, soit par leurs textures ou soit par leurs couleurs, il faut:

- Fusionner ou combiner au moin trois descripteurs de nature différente : un descripteur de forme, un descripteur de texture et un autre descripteur de couleur.
- Combiner les réseaux de neurones (la meilleure approche discriminative de classification) avec les réseaux bayésiens (la meilleure approche générative de classification).
- Améliorer la segmentation d'images.

Ceci, constitue l'objet des expériences 2, 3 et 4. Les résultats seront présentés et analysés pour enfin ressortir la structure finale du système d'annotation automatique d'images le plus performant.

V.2.2.2 Expérience 2 : Fusion ou combinaison des descripteurs

Partant des résultats de l'expérience 1, nous nous sommes intéressés, dans cette expérience, à l'étude de l'effet de la fusion et de la combinaison des descripteurs d'images sur la qualité de l'annotation automatique d'images. Les descripteurs que nous avons utilisés sont les moments de Legendre, les histogrammes de couleurs RGB et les indices d'Haralik basés sur les matrices de cooccurrences chromatiques. Pour la classification, nous avons utilisé les réseaux de neurones et les réseaux bayésiens.

V.2.2.2.1) Fusion des descripteurs

Dans cette approche expérimentale, les descripteurs considérés sont fusionnés en un seul vecteur descripteur. Ce vecteur sera l'entrée de l'un des classificateurs adoptés: les réseaux bayésiens ou les réseaux de neurones.

Le principe de fusion des descripteurs est illustré sur la Figure V-14.

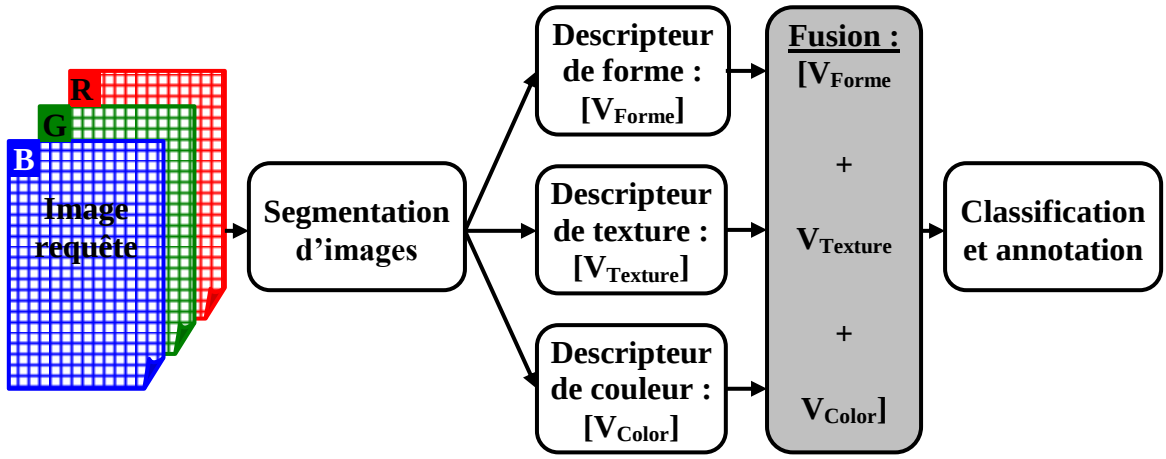


Figure V-14 : Principe de fusion des descripteurs considérés.

Résultats

Les taux d'annotation obtenus, en fusionnant les descripteurs considérés, sont donnés dans le Tableau V-4.

Tableau V-4 : Taux d'annotation obtenus en fusionnant les descripteurs.

Base de donnée	Approche de description	Approche de classification	Taux d'annotation	Taux d'Erreur
ETH-80	Fusion des moments de Legendre, des histogrammes RGB et la Texture	Modèle discriminatif : réseaux de neurones	85.00%	15.00%
	Fusion des moments de Legendre, des histogrammes RGB et la Texture	Modèle génératif : réseaux bayésiens	87.50%	12.50%
COIL-100	Fusion des moments de Legendre, des histogrammes RGB et la Texture	Modèle discriminatif : réseaux de neurones	80.00%	20.00%
	Fusion des moments de Legendre, des histogrammes RGB et la Texture	Modèle génératif : réseaux bayésiens	82.50%	17.50%
NATURE	Fusion des moments de Legendre, des histogrammes RGB et la Texture	Modèle discriminatif : réseaux de neurones	86.67%	13.33%
	Fusion des moments de Legendre, des histogrammes RGB et la Texture	Modèle génératif : réseaux bayésiens	90.00%	10.00%

La matrice de confusion du système d'annotation automatique d'images basé sur la fusion des descripteurs en utilisant les réseaux bayésiens est illustrée sur la Figure V-15 pour des images de la base de données ETH-80.

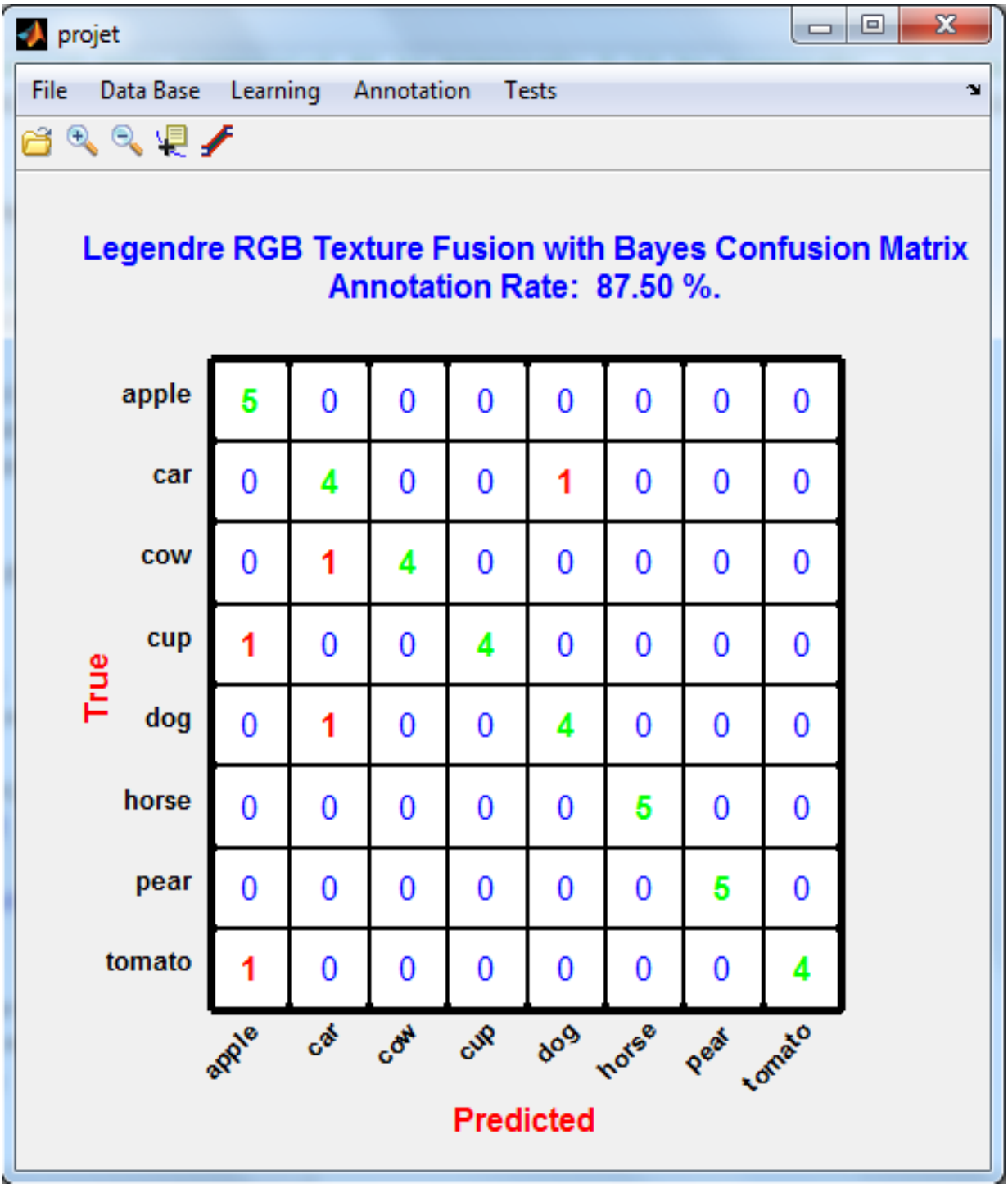


Figure V-15 : Matrice de confusion du système d'annotation automatique d'images basé sur la fusion des descripteurs en utilisant les réseaux bayésiens et la base ETH-80.

La matrice de confusion du système d'annotation automatique d'images basé sur la fusion des descripteurs en utilisant les réseaux bayésiens est illustrée sur la Figure V-16 pour des images de la base de données COIL-100.

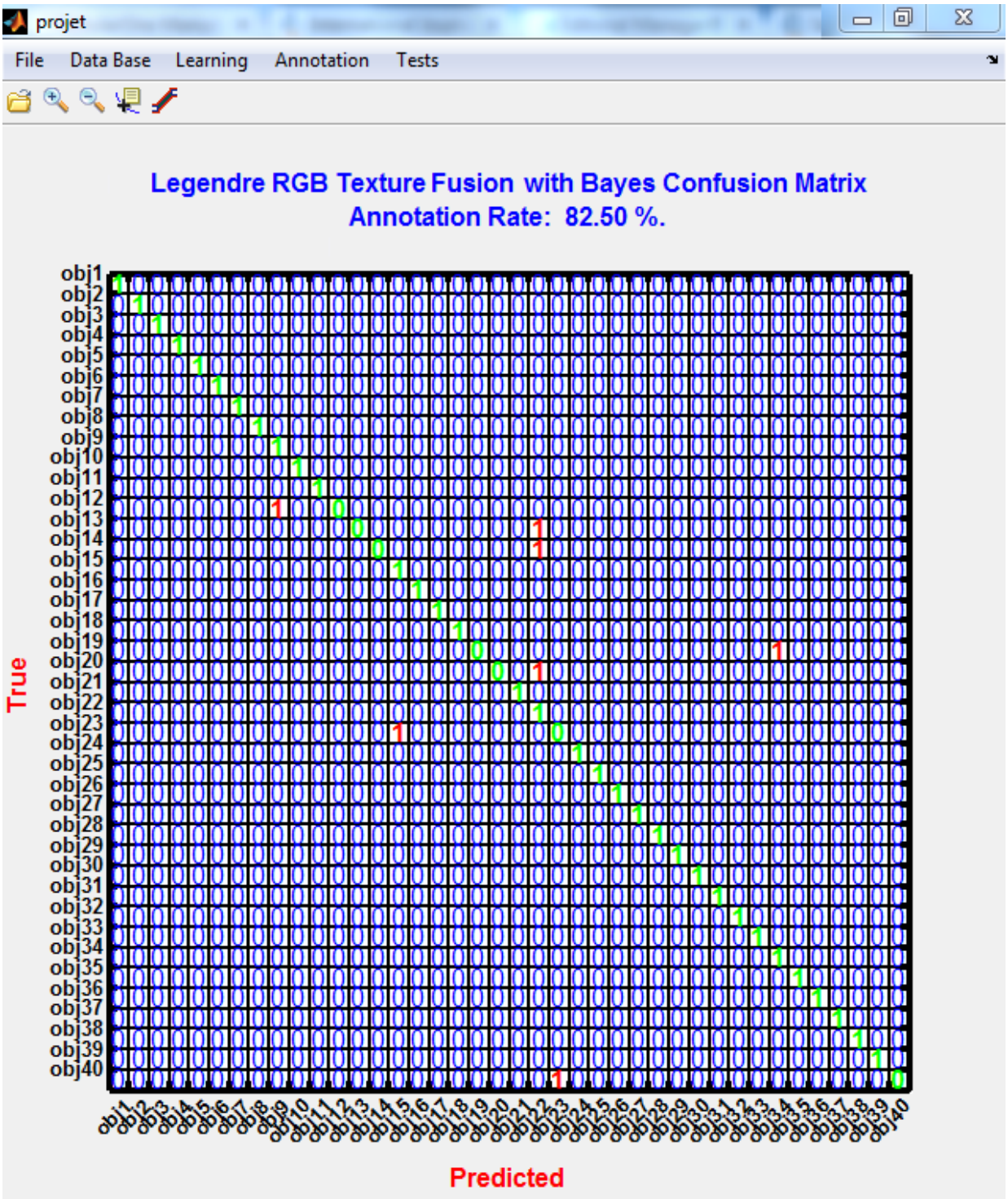


Figure V-16 : Matrice de confusion du système d'annotation automatique d'images basé sur la fusion des descripteurs en utilisant les réseaux bayésiens et la base de données COIL-100.

La matrice de confusion du système d'annotation automatique d'images basé sur la fusion des descripteurs en utilisant les réseaux bayésiens est illustrée sur la Figure V-17 pour des images de la base de données NATURE.

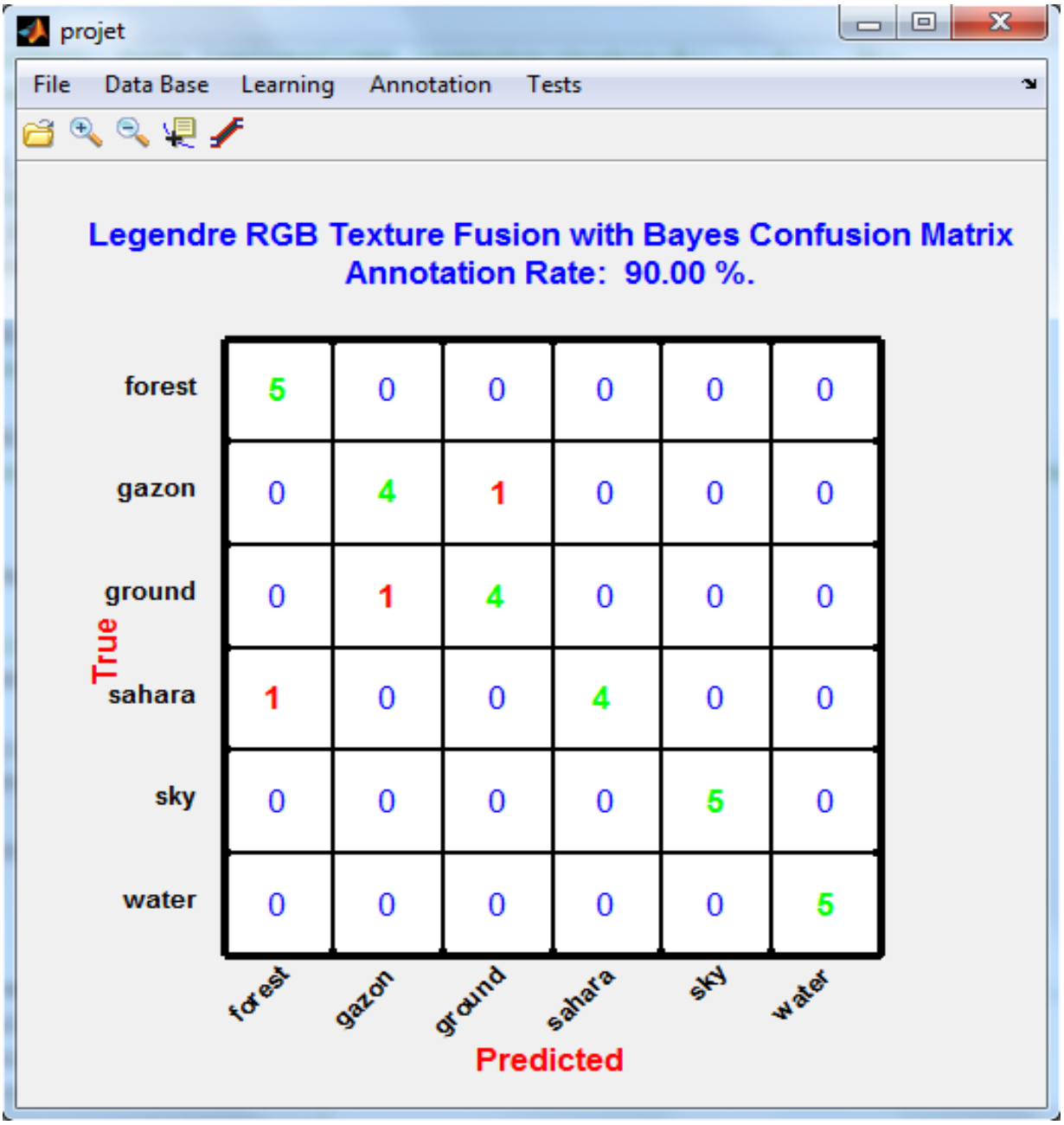


Figure V-17 : Matrice de confusion du système d'annotation automatique d'images basé sur la fusion des descripteurs en utilisant les réseaux bayésiens et la base de données NATURE.

V.2.2.2.2) Combinaison des descripteurs

Dans cette approche expérimentale, les descripteurs sont combinés et utilisés avec l'une des deux approches de classification automatique d'images, à savoir : les réseaux bayésiens et les réseaux de neurones.

Le principe de combinaison des descripteurs est illustré sur la Figure V-18.

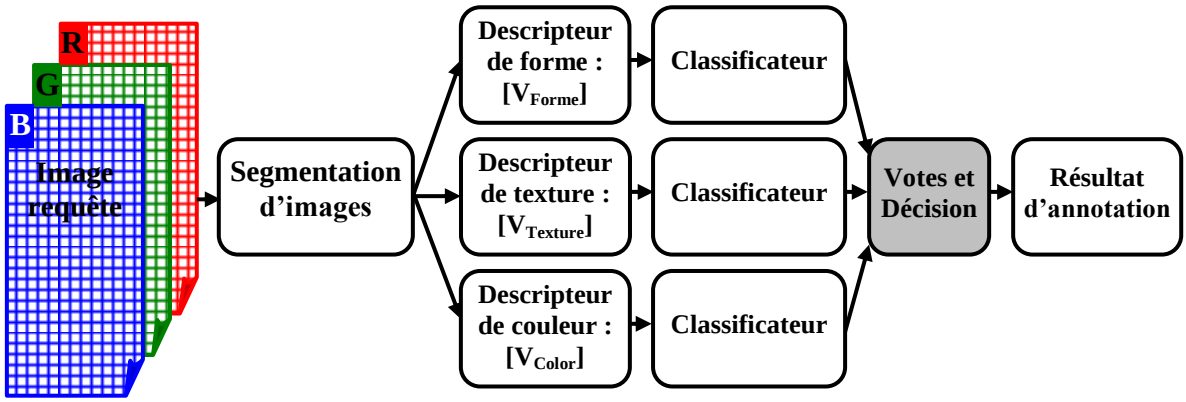


Figure V-18 : Principe de combinaison des descripteurs

Pour cette approche de combinaison, les réseaux bayésiens ou les réseaux de neurones votent pour un mot-clé bien déterminé. La décision est prise en choisissant le mot-clé ayant le maximum de votes.

Résultats

Les résultats d'annotation en combinant les descripteurs sont donnés par le Tableau V-5.

Tableau V-5 : Résultats d'annotation en combinant les descripteurs.

Base de donnée	Approche de description	Approche de classification	Taux d'annotation	Taux d'Erreur
ETH-80	Combinaison des moments de Legendre, des histogrammes RGB et la Texture	Modèle discriminatif : réseaux de neurones	87.50%	12.50%
	Combinaison des moments de Legendre, des histogrammes RGB et la Texture	Modèle génératif : réseaux bayésiens	90.00%	10.00%
COIL-100	Combinaison des moments de Legendre, des histogrammes RGB et la Texture	Modèle discriminatif : réseaux de neurones	82.50%	17.50%
	Combinaison des moments de Legendre, des histogrammes RGB et la Texture	Modèle génératif : réseaux bayésiens	85.00%	15.00%
NATURE	Combinaison des moments de Legendre, des histogrammes RGB et la Texture	Modèle discriminatif : réseaux de neurones	90.00%	10.00%
	Combinaison des moments de Legendre, des histogrammes RGB et la Texture	Modèle génératif : réseaux bayésiens	93.33%	6.77%

La matrice de confusion du système d'annotation automatique d'images basé sur la combinaison des descripteurs en utilisant les réseaux bayésiens est illustrée sur la Figure V-19 pour des images de la base de données ETH-80.

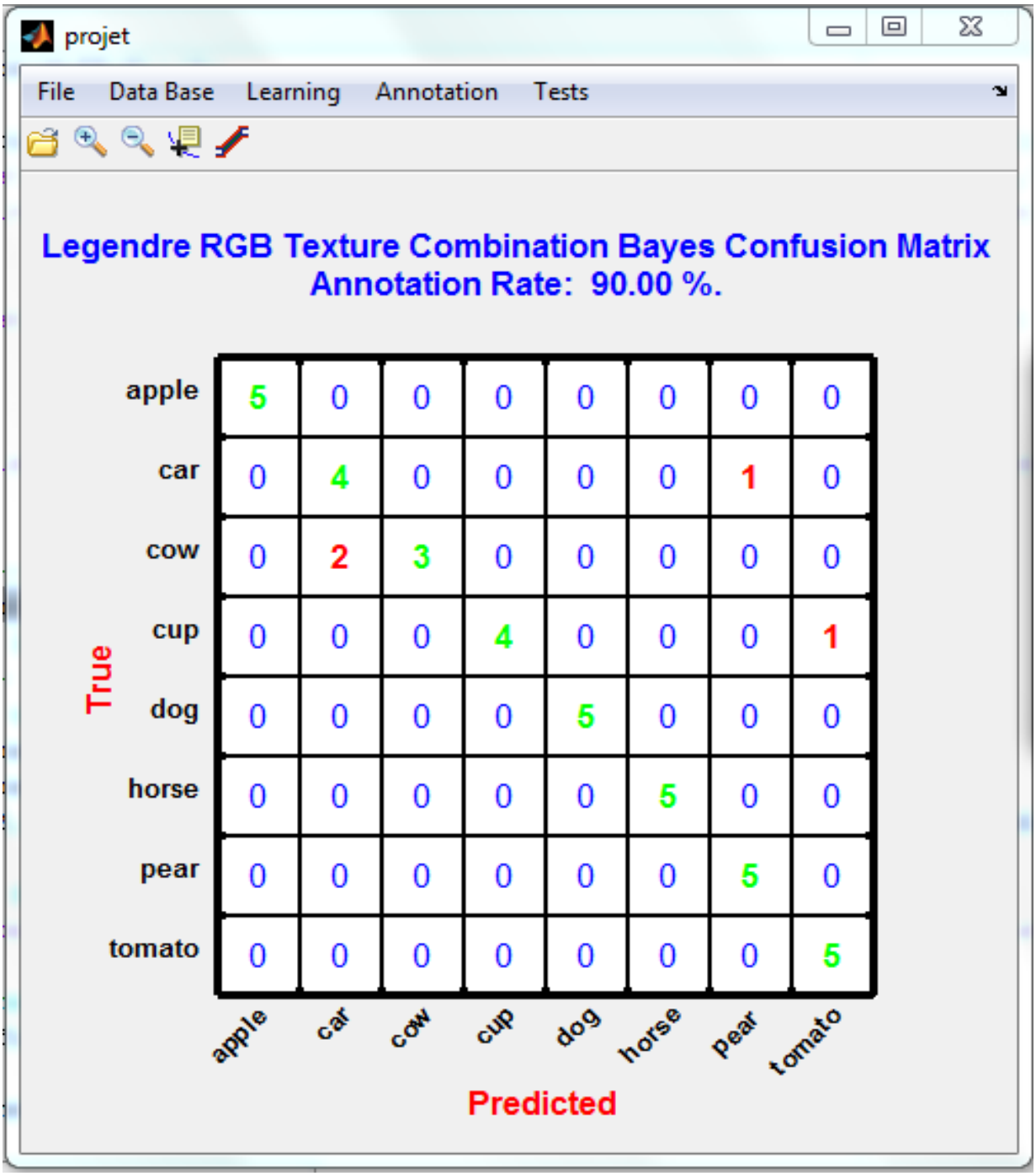


Figure V-19 : Matrice de confusion du système d'annotation automatique d'images basé sur la combinaison des descripteurs en utilisant les réseaux bayésiens et la base de données ETH-80.

La matrice de confusion du système d'annotation automatique d'images basé sur la combinaison des descripteurs en utilisant les réseaux bayésiens est illustrée sur la Figure V-20 pour des images de la base de données COIL-100.

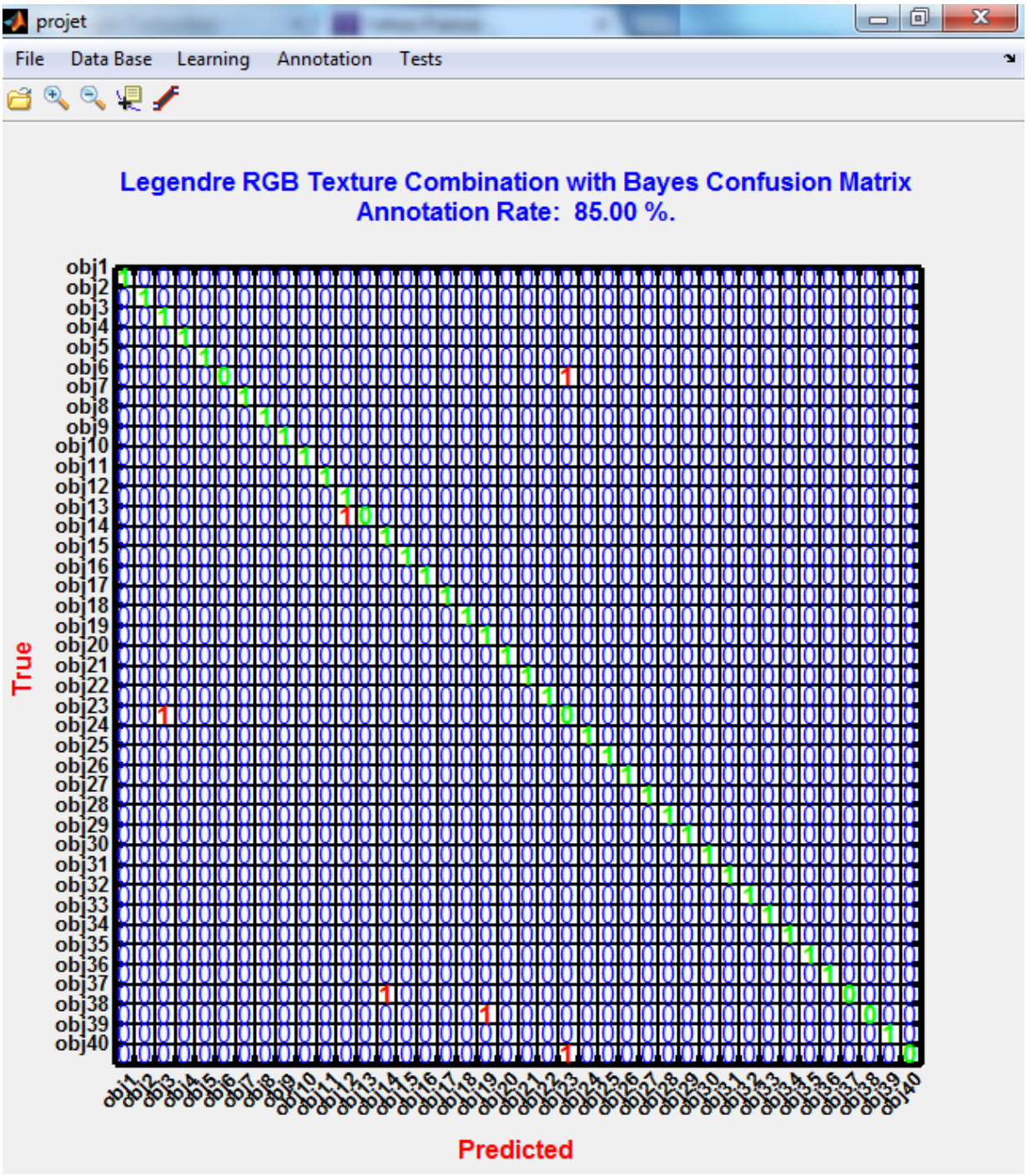


Figure V-20 : Matrice de confusion du système d'annotation automatique d'images basé sur la combinaison des descripteurs en utilisant les réseaux bayésiens et la base de données COIL-100.

La matrice de confusion du système d'annotation automatique d'images basé sur la combinaison des descripteurs en utilisant les réseaux bayésiens est illustrée sur la Figure V-21 pour des images de la base de données NATURE.

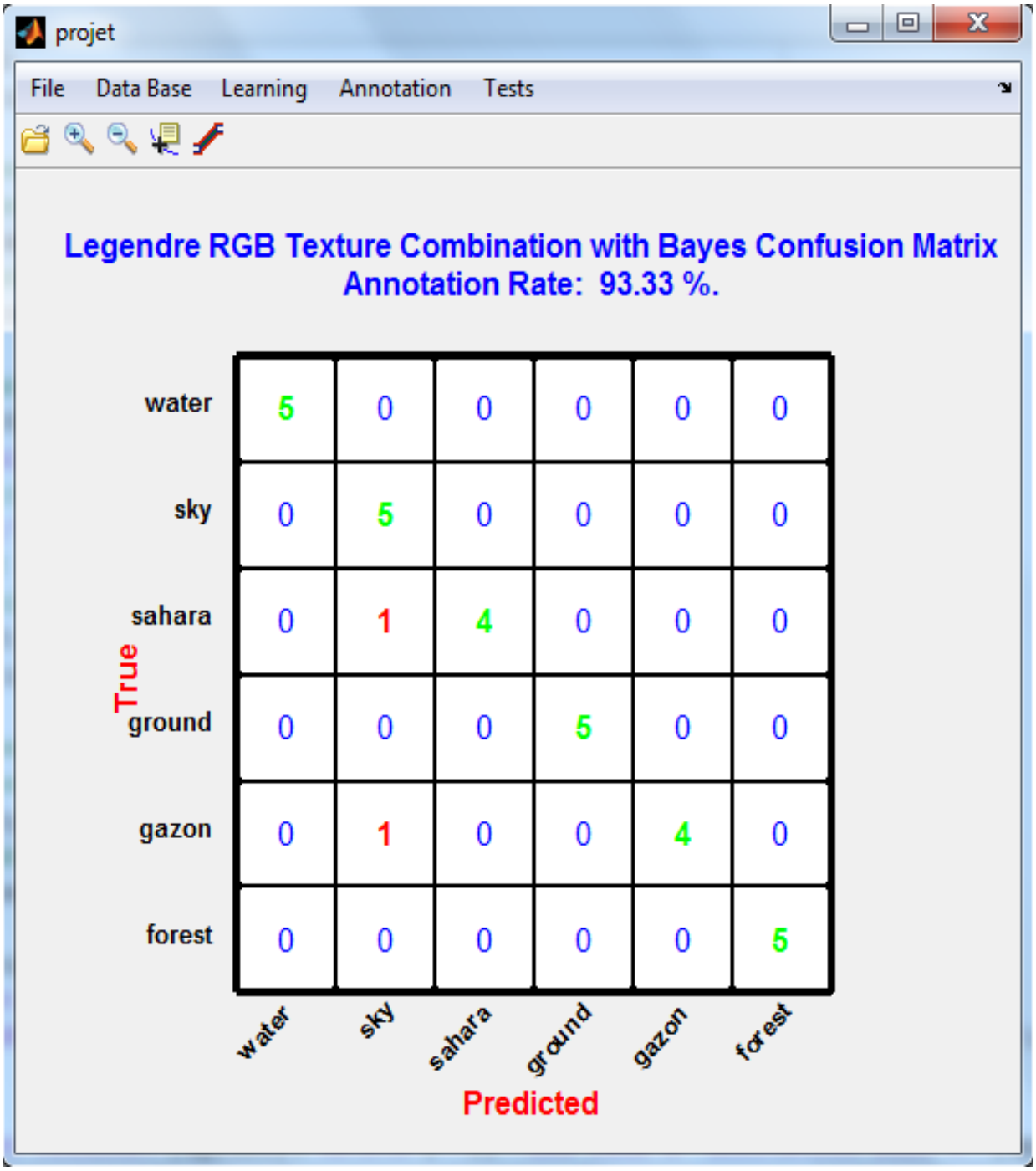


Figure V-21 : Matrice de confusion du système d'annotation automatique d'images basé sur la combinaison des descripteurs en utilisant les réseaux bayésiens et la base de données NATURE.

V.2.2.2.3) Analyse des résultats et conclusion

L'examen des résultats donnés par les Tableaux V-4 et V-5 et les Figures V-15, V-16, V-17, V-18, V-19, V-20 et V-21, montre que :

- La fusion ou la combinaison des descripteurs permet d'améliorer le taux d'annotation d'images pour chacun des classificateurs utilisés. Ce résultat a été prévisible, car les classificateurs ont saisi plus d'informations pour distinguer les objets et par la suite les classifier.

- Le taux d'annotation obtenu par la combinaison des descripteurs est meilleur que celui obtenu par la fusion. Ce taux peut être amélioré davantage en augmentant le nombre de descripteurs combinés. Dans le cas de la fusion, cela n'est pas évident, car la taille du vecteur des descripteurs fusionnés devient très grande.

D'après ces remarques, il apparait que la combinaison des descripteurs est la structure la plus performante pour étudier la combinaison des réseaux de neurones avec les réseaux bayésiens qu'on a signalé dans l'expérience 1. Ceci fera l'objet de l'experience 3.

V.2.2.3 Expérience 3 : Combinaison des classificateurs et descripteurs

Partant des remarques dégagées au niveau des deux expériences précédentes, nous avons combiné, en plus des descripteurs, les réseaux de neurones et les réseaux bayésiens afin de tirer bénéfice de la complémentarité de ces deux approches de classification (discriminative et générative). Le principe de cette combinaison est illustré par le schéma bloc représenté par la Figure V-22. Ainsi, avec la combinaison des 3 types de descripteurs (les descripteurs de formes : moments de Legendre, les descripteurs de couleurs : histogrammes de couleurs RGB et les descripteurs de texture : indices d'Haralik basés sur les matrices de cooccurrences chromatiques) et des 2 types de classificateurs considérés, on aura un maximum de vote égal à 3 x 2 = 6. Chaque classificateur avec chaque descripteur vote pour un mot-clé donné. Le mot-clé ayant un maximum de votes sera considéré comme le mot convenable pour l'annotation d'un objet contenu dans une image requête.

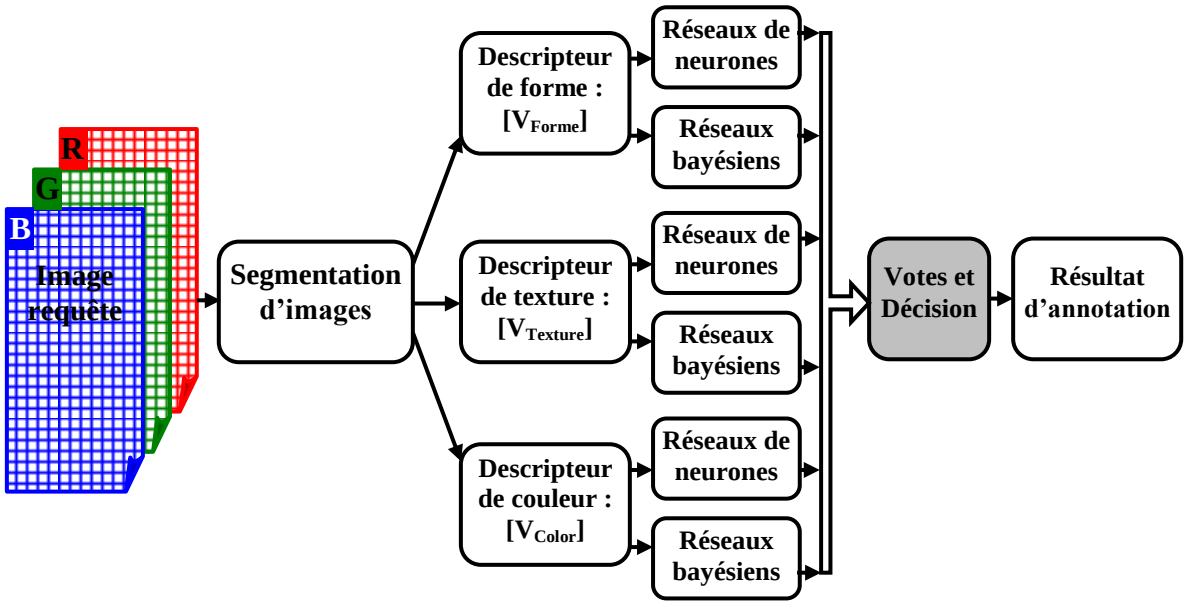


Figure V-22 : Schéma bloc du système d'annotation automatique d'images basé sur la combinaison des classificateurs et descripteurs.

V.2.2.3.1) Résultats

Le Tableau V-6 donne les taux d'annotation de la combinaison des descripteurs et des classificateurs, réseaux de neurones et réseaux bayésiens, en utilisant les bases de données d'images NATURE, ETH-80 et COIL-100.

Tableau V-6 : Résultats de l'approche de combinaison des descripteurs et classificateurs en utilisant les bases de données d'images NATURE, ETH-80, et COIL-100.

Base de donnée	Approche de description	Approche de classification	Taux d'annotation	Taux d'Erreur
ETH-80	Combinaison des moments de Legendre, des histogrammes RGB et la Texture	Modèle discriminatif et génératif : réseaux de neurones et réseaux bayésiens	92.50%	7.50%
COIL-100	Combinaison des moments de Legendre, des histogrammes RGB et la Texture	Modèle discriminatif et génératif : réseaux de neurones et réseaux bayésiens	87.50%	12.50%
NATURE	Combinaison des moments de Legendre, des histogrammes RGB et la Texture	Modèle discriminatif et génératif : réseaux de neurones et réseaux bayésiens	96.67%	3.33%

La Figure V-23 donne la matrice de confusion, pour des images de la base de données ETH-80, en combinant les descripteurs et les classificateurs adoptés.

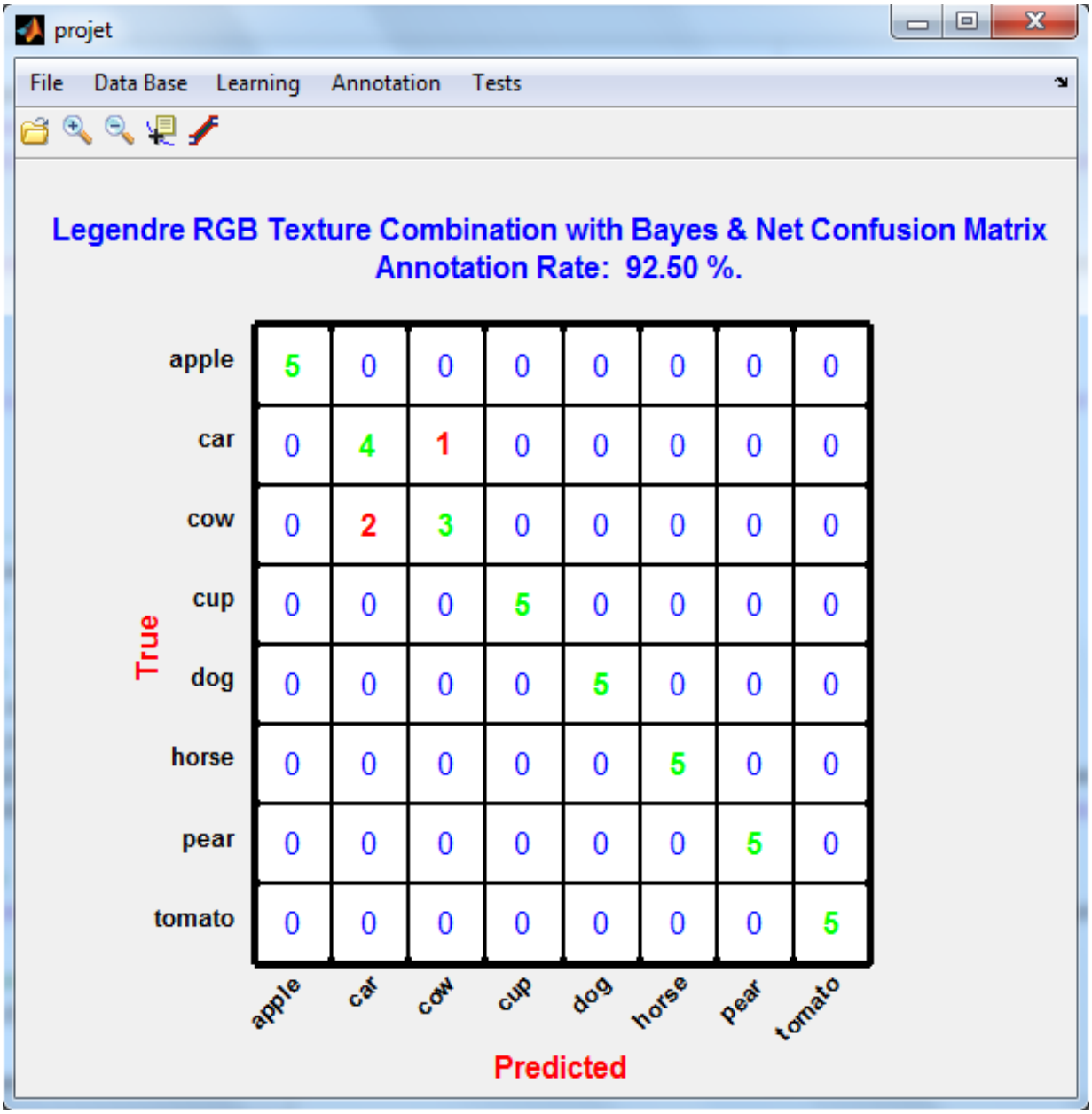


Figure V-23 : Matrice de confusion dans le cas de la combinaison des descripteurs et classificateurs pour des images de la base de données ETH-80.

La Figure V-24 donne la matrice de confusion, pour des images de la base de données COIL-100, en combinant les descripteurs et les classificateurs adoptés.

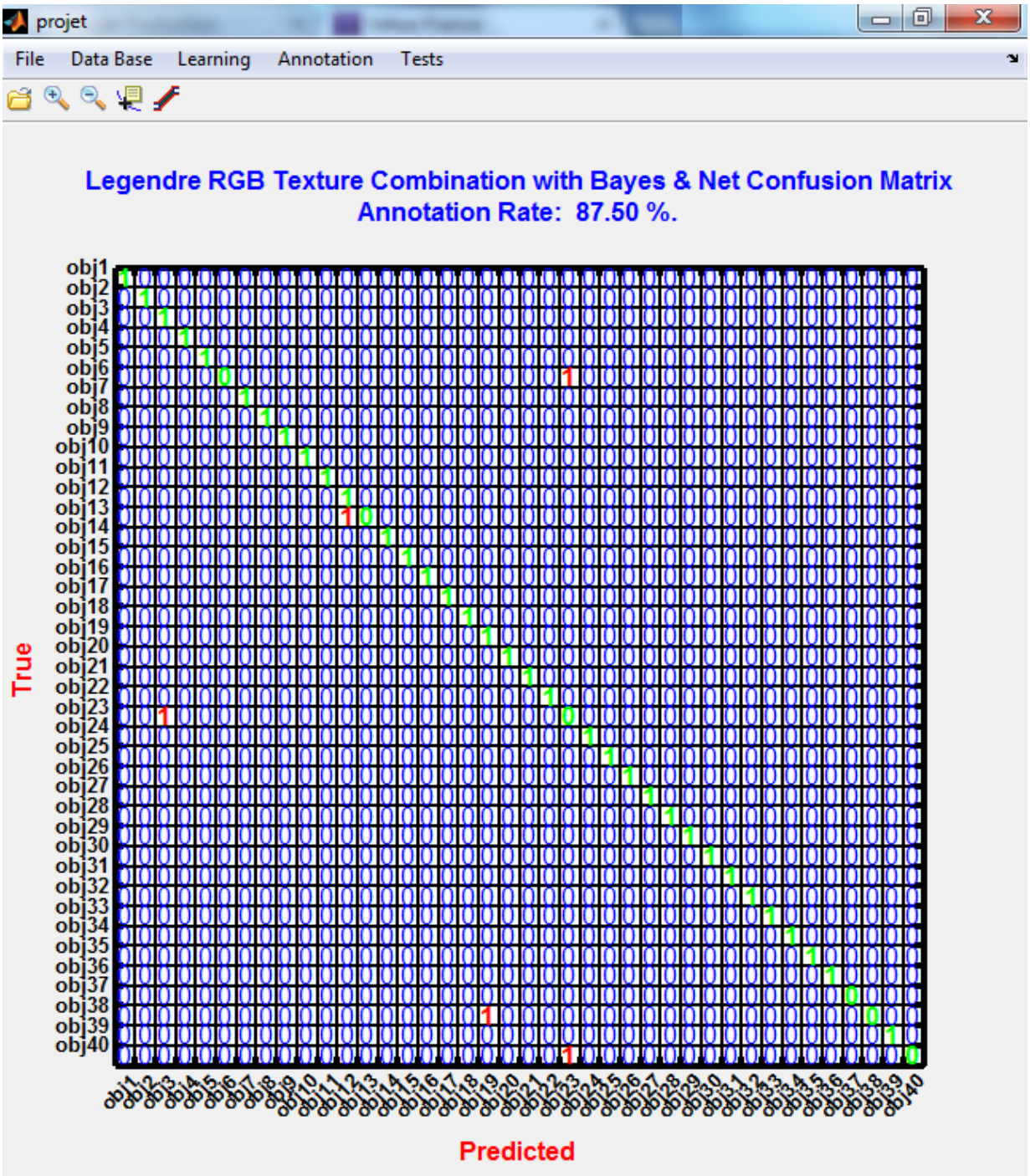


Figure V-24 : Matrice de confusion dans le cas de la combinaison des descripteurs et classificateurs pour des images de la base de données COIL-100.

La Figure V-25 donne la matrice de confusion, pour des images de la base de données NATURE, en combinant les descripteurs et les classificateurs adoptés.

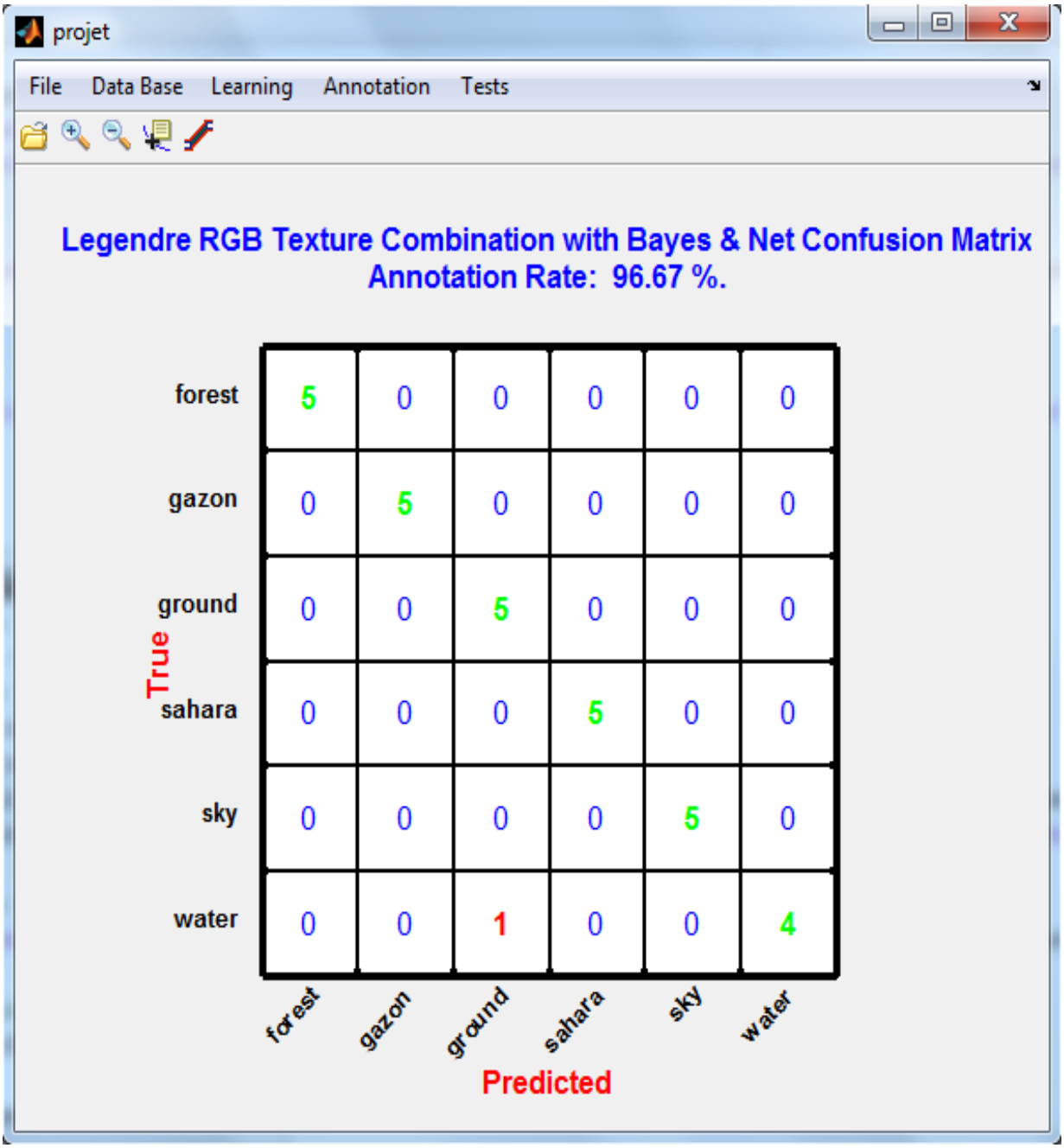


Figure V-25 : Matrice de confusion dans le cas de la combinaison des descripteurs et classificateurs, pour des images de la base de données NATURE.

V.2.2.3.2) Analyse des résultats et conclusion

Nous constatons, d'après les résultats présentés par le Tableau V-6, les Figures V-23, V-24 et V-25, que la combinaison des descripteurs considérés et des réseaux de neurones avec les réseaux bayésiens, a permis d'améliorer nettement la qualité de l'annotation d'images. Ceci nous permet de conclure que pour avoir un système d'annotation automatique d'images performant, il faut exploiter la complémentarité des deux approches discriminative et générative de la classification. Ainsi, nous avons adopté pour le système d'annotation à

réaliser, une architecture qui consiste à combiner, en plus des descripteurs, les réseaux de neurones avec les réseaux bayésiens.

Pour rendre cette architecture plus performante, nous nous sommes intéressés au regroupement des régions adjacentes afin d'avoir des objets sémantiquement compacts en vue de faciliter leur classification et augmenter ainsi le taux d'annotation automatique d'images. Ceci fera l'objet de l'expérience 4.

V.2.2.4 Expérience 4 : Regroupement des régions adjacentes de l'image

Il est très clair que les régions résultantes d'une segmentation automatique d'images ne représentent pas tout à fait les objets réels contenus dans les images. En effet, les méthodes de segmentation automatiques d'images utilisent souvent des agrégats et prédicats de bas niveau qui ne permettent pas de garder la compacité des objets composés de plusieurs couleurs. Ainsi, nous avons, sur la Figure V-26, la segmentation automatique d'une image représentant l'objet « car » qui est segmenté en plusieurs clusters. Elle montre aussi la possibilité de regroupement de ces clusters ou régions pour avoir un objet sémantiquement compact. Nous pouvons voir aussi à partir de cette figure que le regroupement des clusters 1 et 3 représente un objet compact pouvant être annoté facilement de façon plus correcte que les objets des autres clusters non regroupés. D'où l'intérêt majeur de regroupement et fusion des régions adjacentes pour l'annotation automatique d'images.

Ainsi dans cette expérience, le regroupement des régions avant leur annotation peut aider à résoudre ce problème. A partir d'un ensemble de régions, plusieurs combinaisons de regroupement sont possibles. Tous les regroupements possibles sont classés par ordre de probabilité d'annotation résultante de la combinaison de plusieurs classificateurs et plusieurs descripteurs. Les regroupements ayant les plus grandes probabilités d'annotation, représentent probablement des objets compacts dans l'image, et ils sont gardés dans la liste des mots-clés d'annotation du fait qu'ils constituent les meilleurs candidats pour l'annotation automatique de l'image requête.

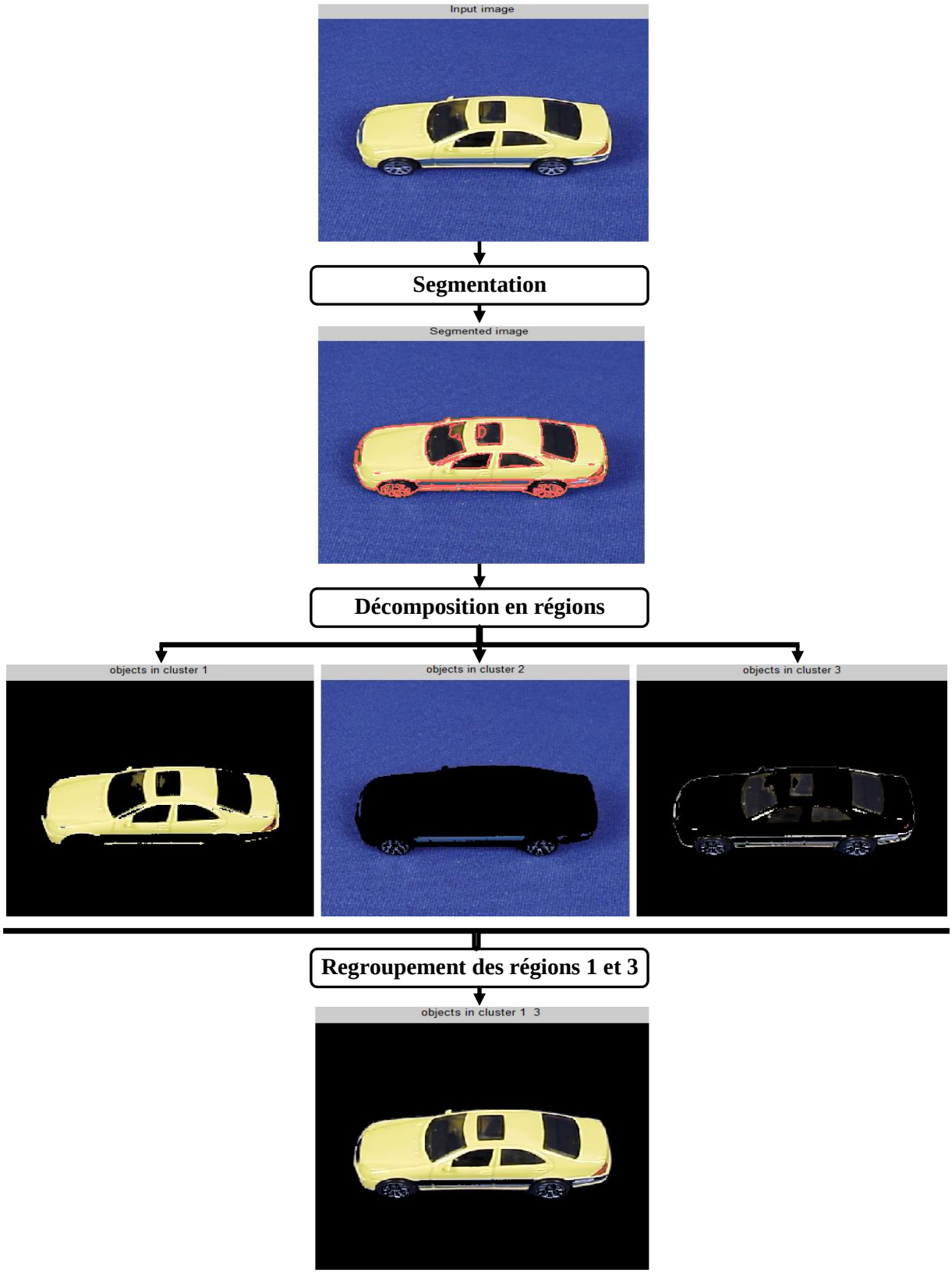


Figure V-26 : Exemple de segmentation et regroupement des clusters d'une image.

La Figure V-27 illustre le principe utilisé pour le regroupement des régions adjacentes ainsi que la structure finale adoptée pour le système d'annotation automatique d'images.

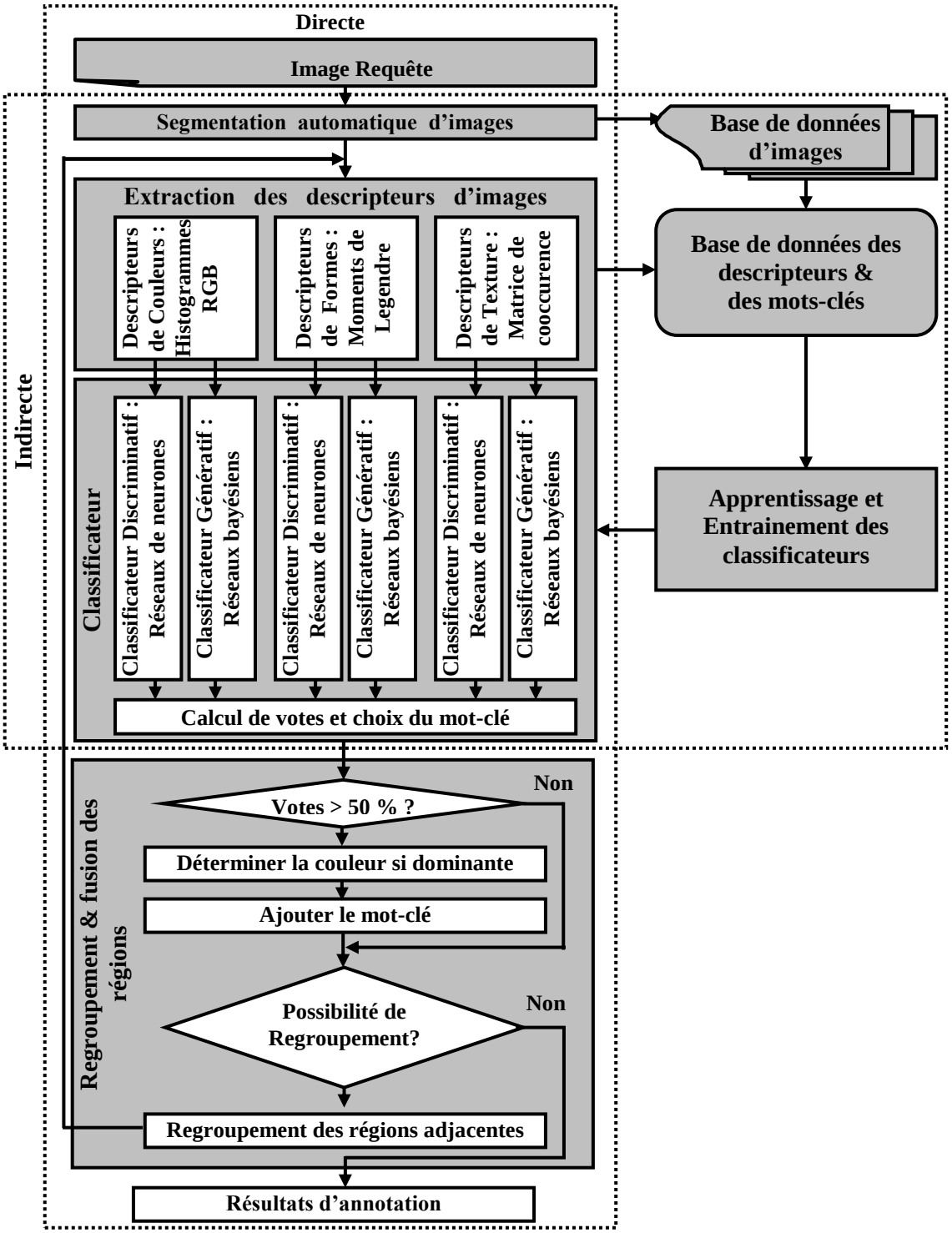


Figure V-27 : Structure finale du système d'annotation automatique d'images basé sur le regroupement des régions adjacentes et la combinaison des descripteurs et classificateurs.

V.2.2.4.1) Résultats

Le Tableau V-7 montre les résultats obtenus en combinant les descripteurs et les classificateurs, et en utilisant aussi le regroupement des régions adjacentes pour des images de la base de données ETH-80 et COIL-100.

Tableau V-7 : Résultats obtenu en utilisant le regroupement des régions adjacentes et la combinaison des descripteurs et classificateurs pour des images de la base de données ETH-80 et COIL-100.

Base de donnée	Taux d'annotation	Taux d'Erreur
ETH-80	97.50%	2.50%
COIL-100	92.50%	7.50%

La matrice de confusion dans le cas d'utilisation du regroupement des régions adjacentes et la combinaison des descripteurs et classificateurs pour des images de la base ETH-80, est illustrée sur la Figure V-28.

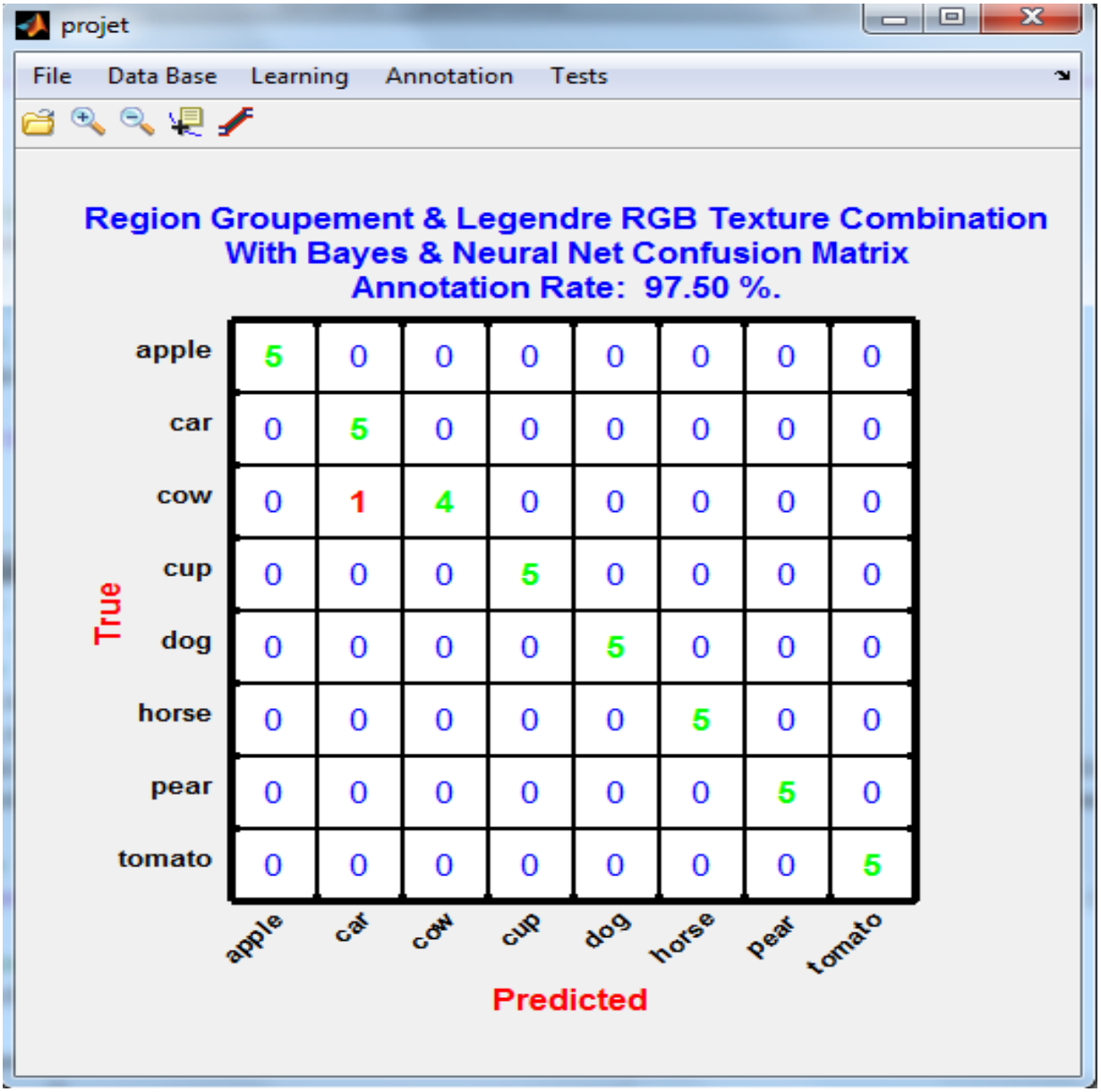


Figure V-28 : Matrice de confusion dans le cas d'utilisation du regroupement des régions adjacentes et la combinaison des descripteurs et classificateurs pour ETH-80.

La Figure V-29 illustre la matrice de confusion dans le cas d'utilisation du regroupement des régions adjacentes et la combinaison des descripteurs et classificateurs pour des images de la base de données COIL-100.

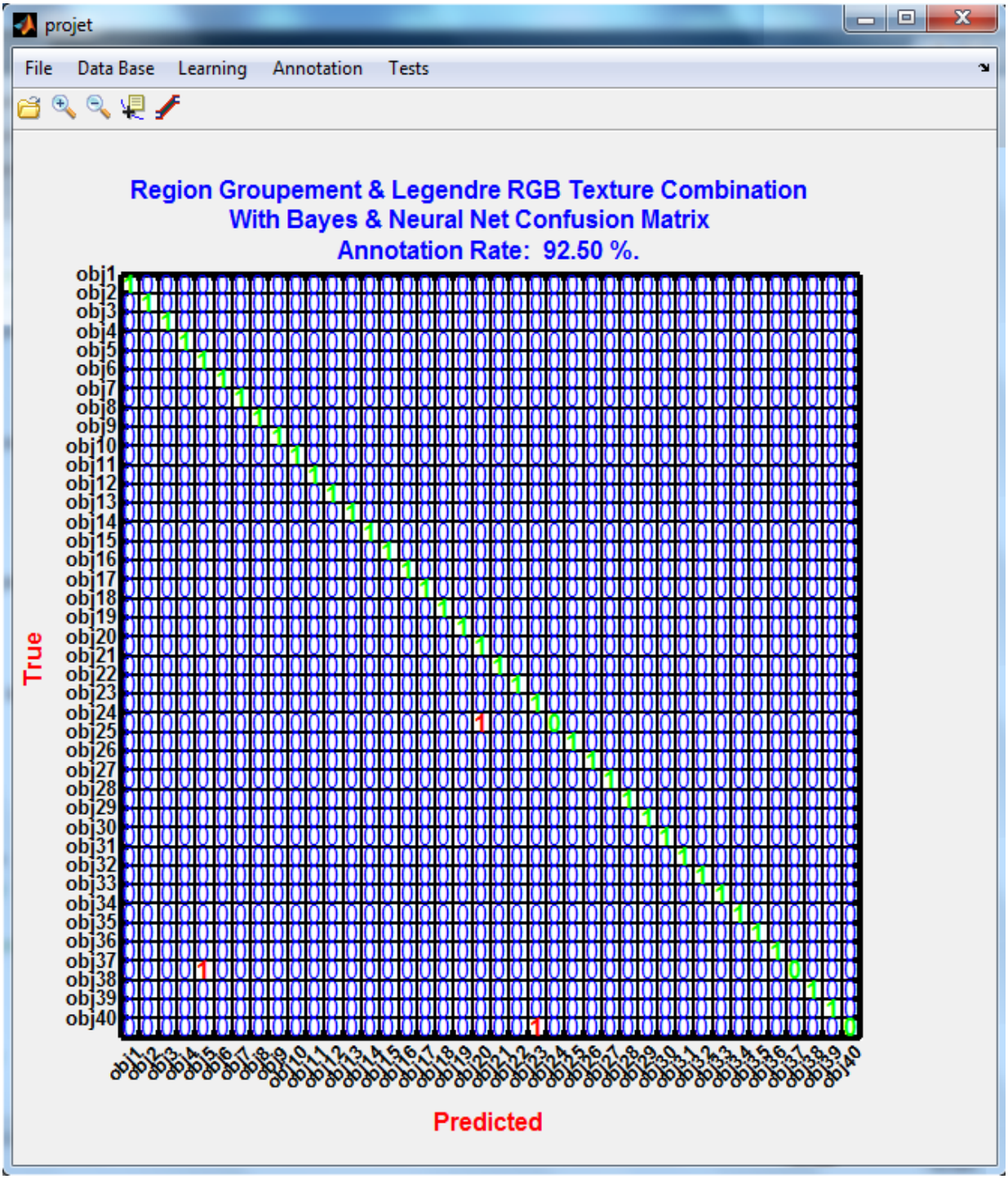


Figure V-29 : Matrice de confusion dans le cas d'utilisation du regroupement des régions adjacentes et la combinaison des descripteurs et classificateurs pour des images de la base de données COIL-100.

V.2.2.4.2) Analyse des résultats et conclusion

Les résultats obtenus, présentés par le Tableau V-7 et les Figures V-28 et V-29, montrent l'utilité de regroupement des régions adjacentes pour augmenter le taux du système d'annotation automatique d'images. Nous pouvons voir à partir de la Figure V-28 que les problèmes d'annotation de l'objet « car » ont été réglés. Ceci montre bien l'efficacité du regroupement de régions adjacentes lors du processus d'annotation automatique d'images. En effet, ce regroupement permet d'avoir des régions, représentant les objets contenus dans les images, compactes et pouvant être classifiées et annotées correctement d'une manière plus efficace que l'annotation qui résulte directement de la segmentation.

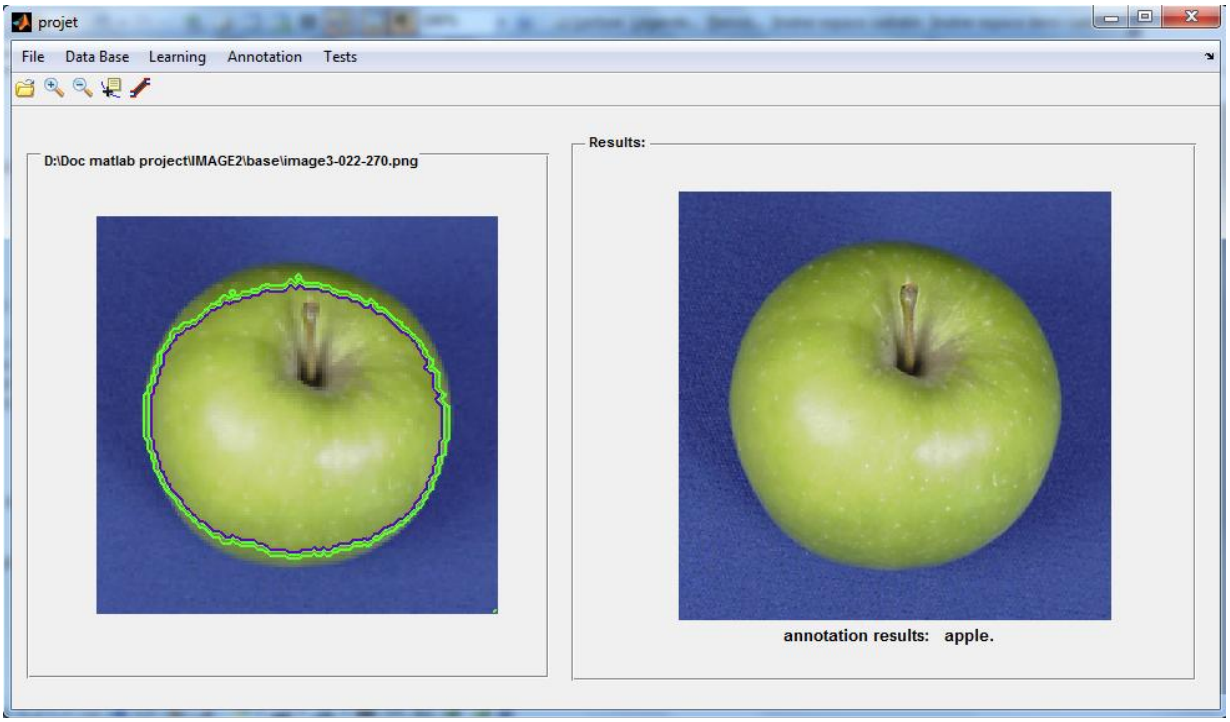
La structure finale et l'architecture adoptée pour le système d'annotation automatique d'images est représentée par la Figure V-27. Des exemples d'annotation automatique d'images sont présentés dans le paragraphe suivant.

V.3 Evaluation

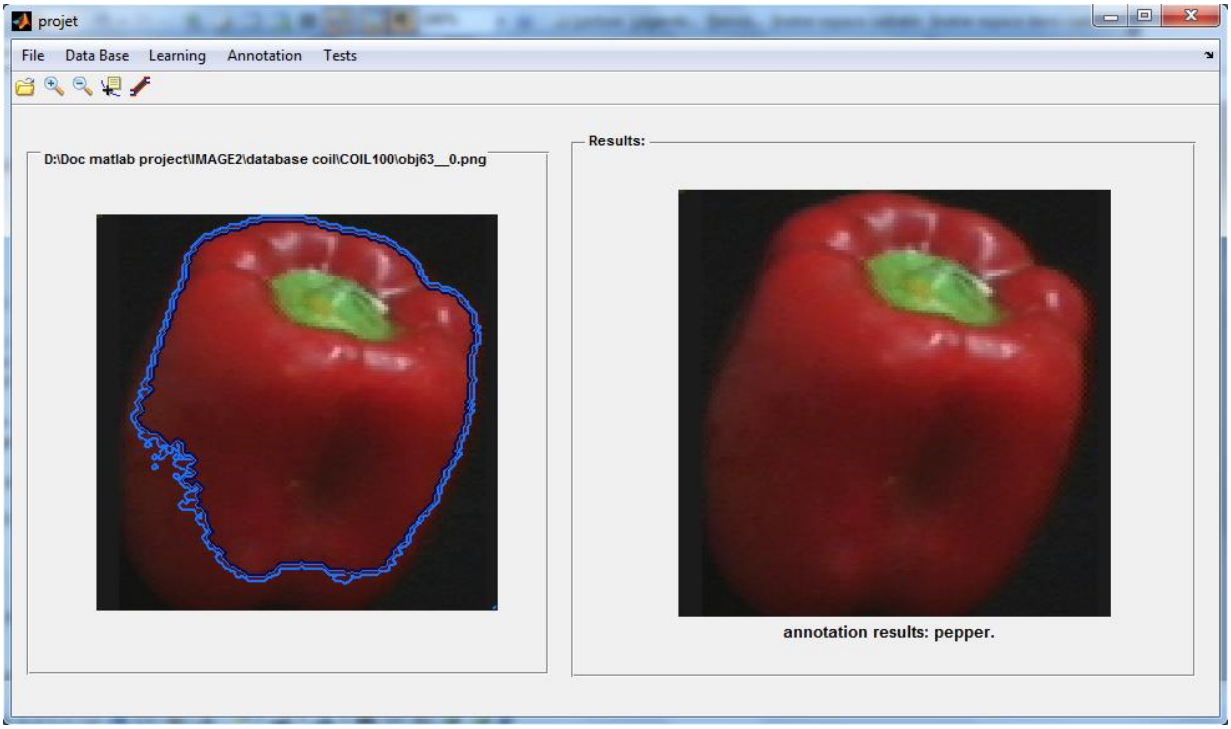
Dans cette partie, nous nous sommes intéressés à l'évaluation du système d'annotation automatique d'images que nous avons réalisé. Pour cette fin, nous l'avons utilisé pour annoter de nouvelles images, plus ou moins complexes, contenant des objets appartenant aux trois bases utilisées au cours de l'apprentissage. Certains résultats d'annotation obtenus, sont présentés par les Figures V-30, V-31 et V-32 en utilisant l'interface graphique. Nous constatons que les objets qui se trouvent dans les images de la Figure V-30, sont bien annotés par leurs mots clés, et que ces mêmes images sont annotées, en plus de leurs mots clés, par leurs couleurs dominantes (Figure V-31). En ce qui concerne l'image de la Figure V-32, nous remarquons que l'objet « cow » est mal annoté car il est mal segmenté. Certes, pour annoter une image contenant plus qu'un objet, il faut la segmenter correctement afin d'extraire les objets à annoter. Ceci n'est pas toujours évident. En effet, il n'y a pas à présent de méthode universelle pour segmenter correctement les images. C'est pourquoi pour évaluer les performances des systèmes d'annotation automatique d'images de l'état de l'art, il faut utiliser uniquement des images pré segmentées.

Ainsi, d'après ces remarques et les résultats d'annotation obtenus, nous pouvons conclure que le système que nous avons développé permet de réaliser de bonnes performances en termes de réduction du fossé sémantique.

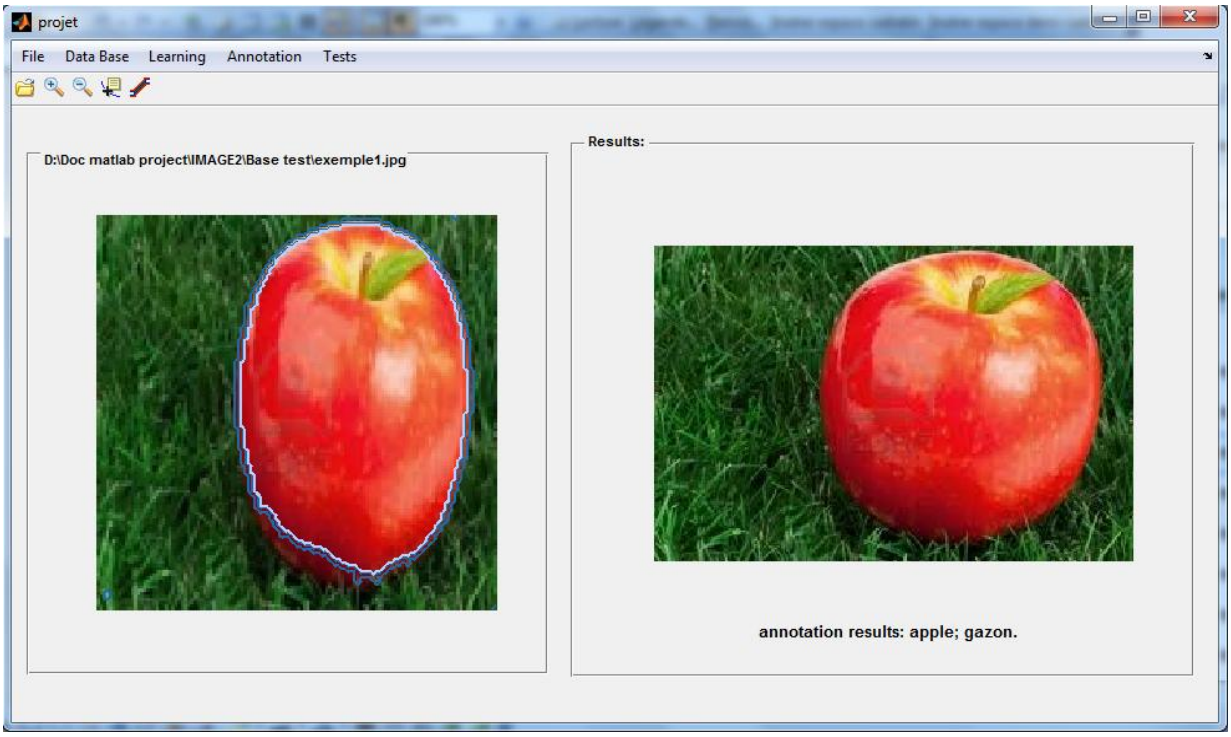
Il faut noter que nous n'avons pas comparé les performances du système d'annotation automatique d'images que nous avons réalisé avec celles d'autres systèmes de l'état de l'art, car les corpus d'images, sur lesquels les expériences sont réalisées, sont différents.



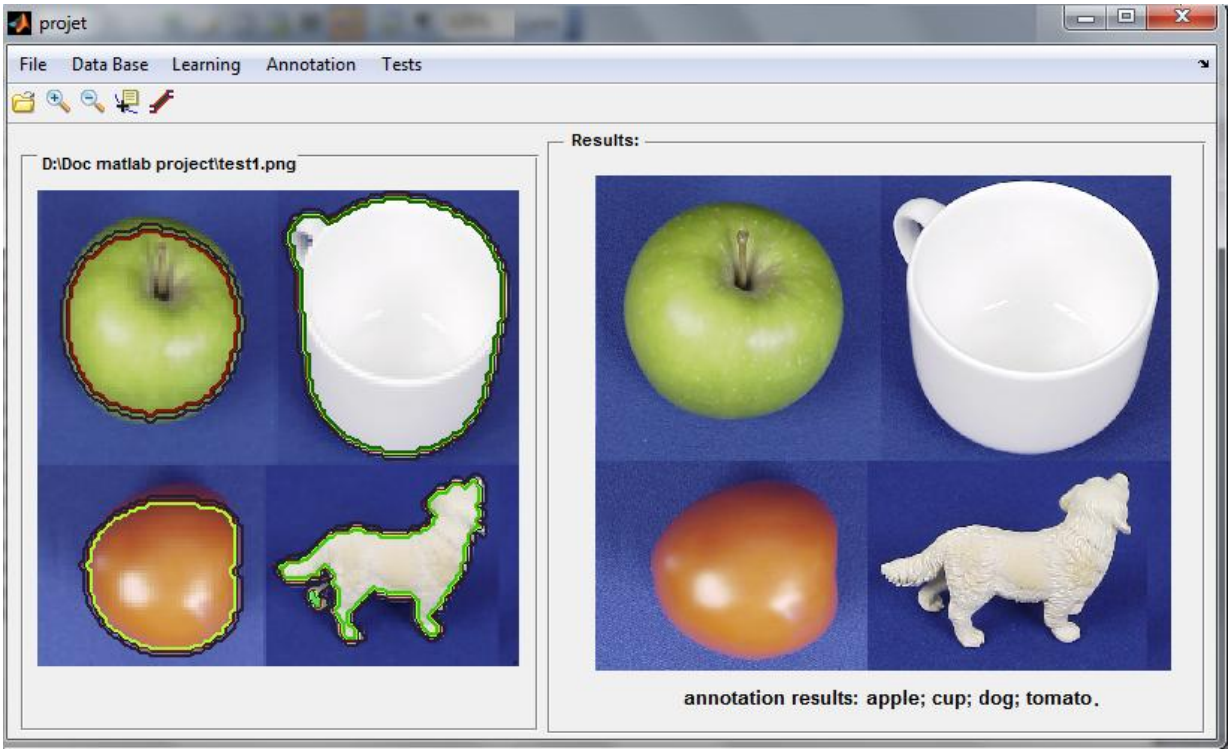
a)



b)

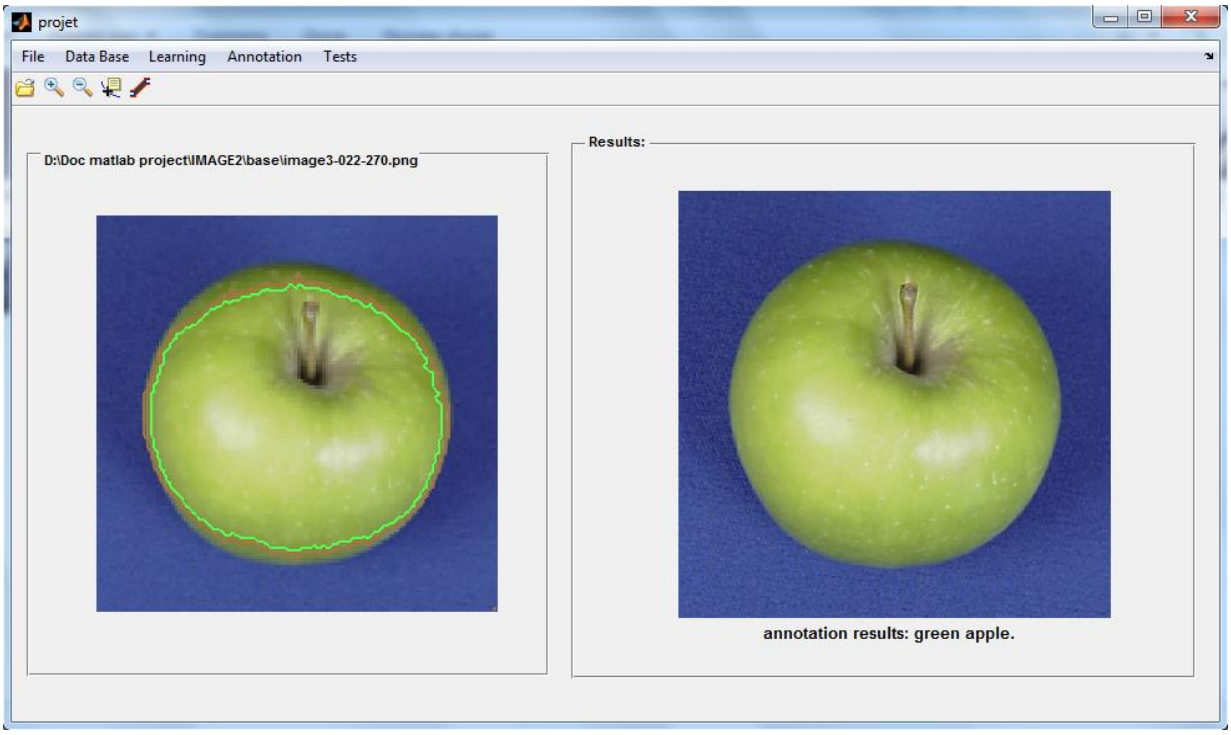


c)

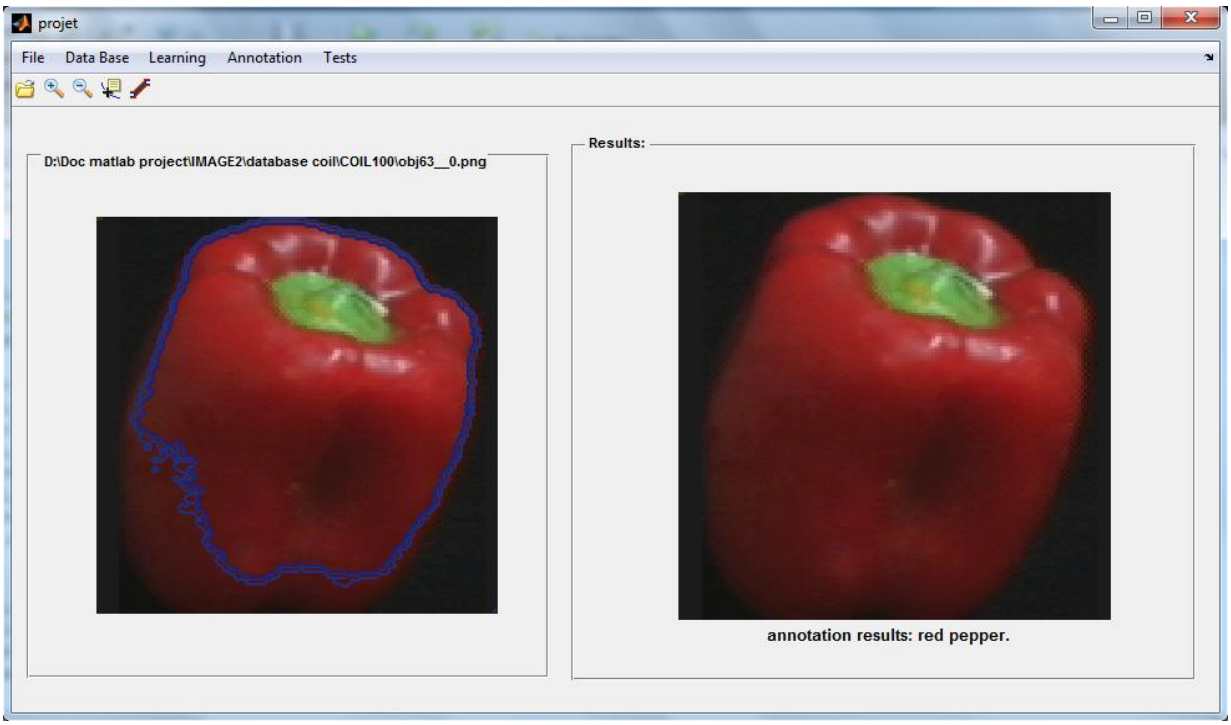


d)

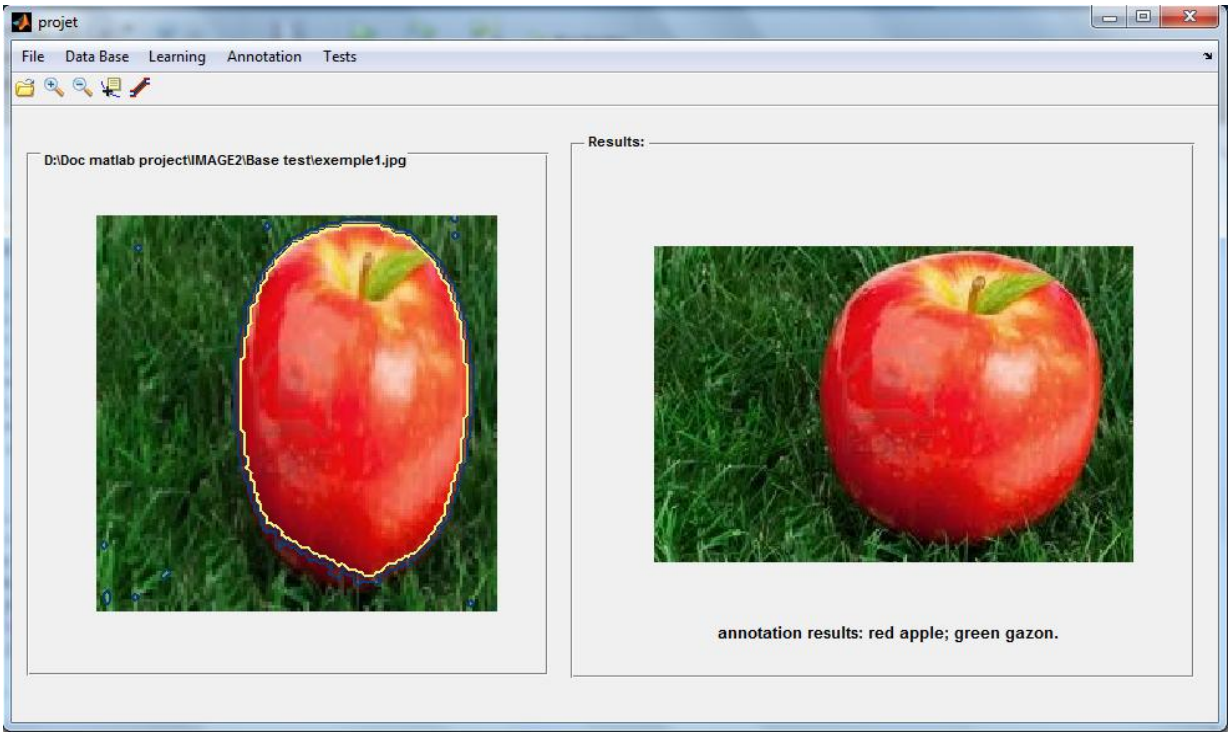
Figure V-30 : Exemple de résultats d'annotation d'images sans couleurs dominantes.


projet
File Data Base Learning Annotation Tests
D:\Doc matlab project\IMAGE2\base\image3-022-270.png
Results:
annotation results: green apple.

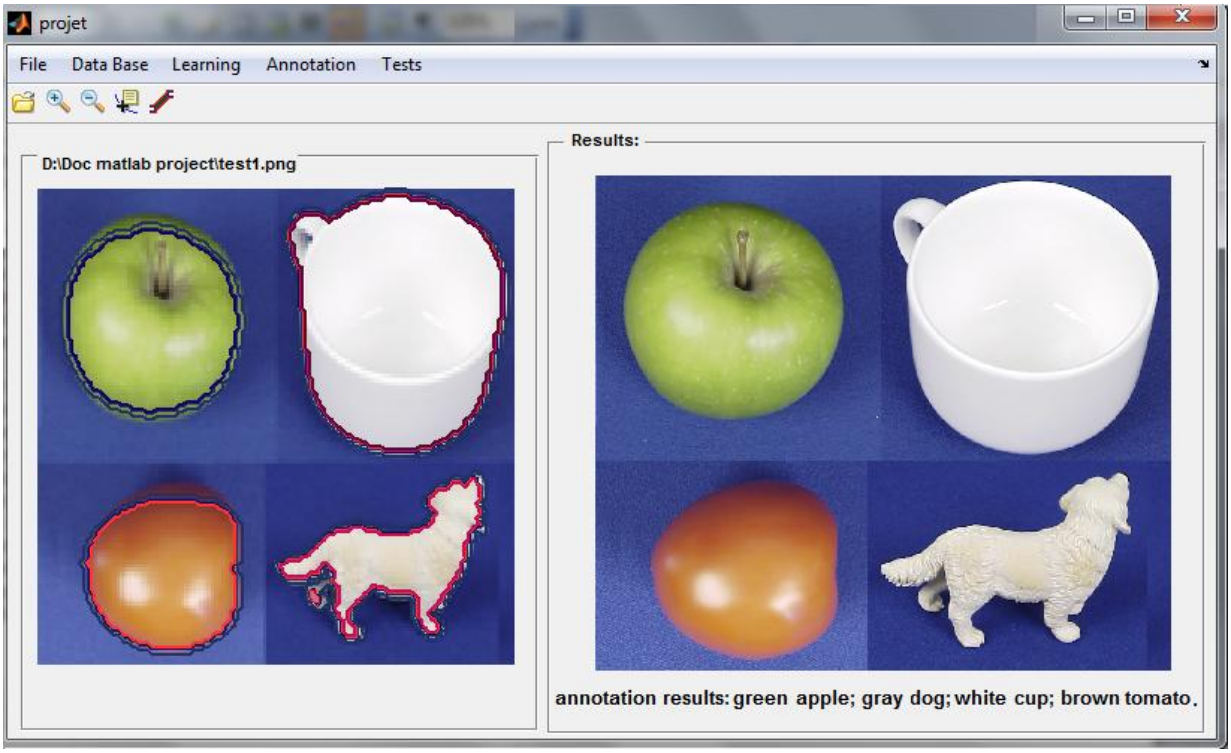
a)


projet
File Data Base Learning Annotation Tests
D:\Doc matlab project\IMAGE2\database coil\COIL100\obj63__0.png
Results:
annotation results: red pepper.

b)



c)



d)

Figure V-31 : Exemple de résultats d'annotation d'images avec couleurs dominantes.

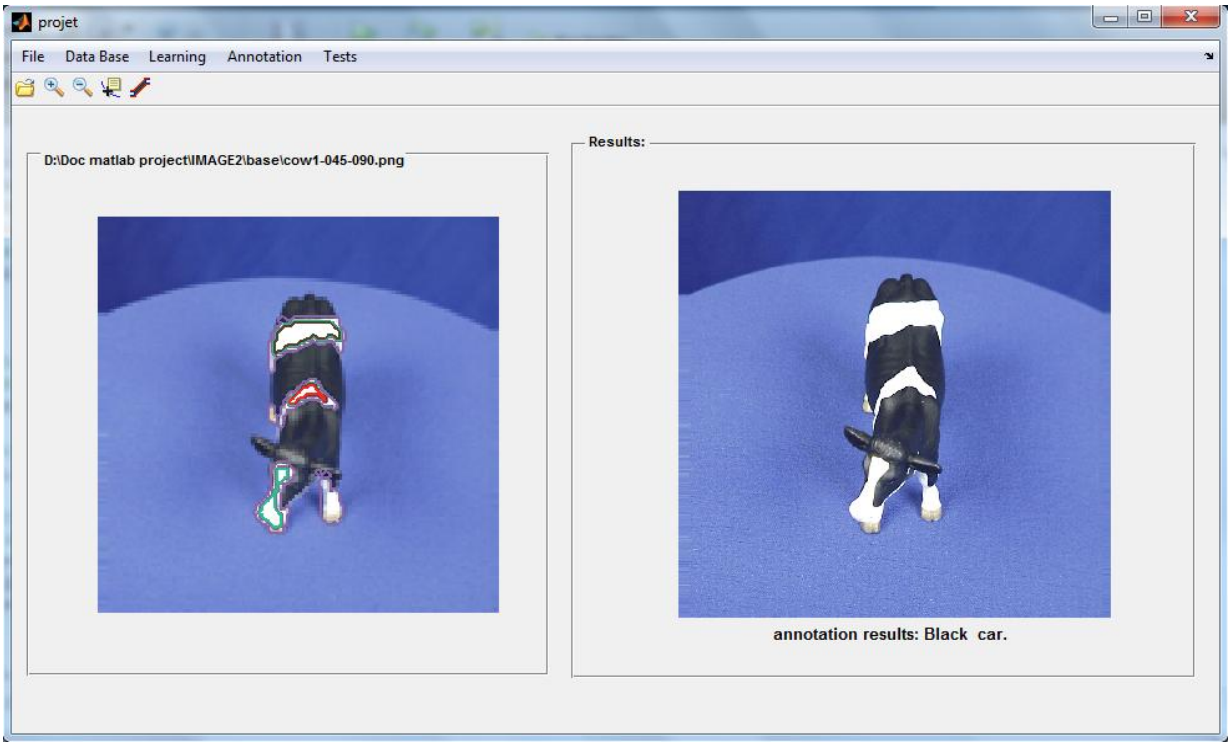


Figure V-32 : Exemple de faux résultat obtenu à l'aide du système d'annotations proposé.

V.4 Conclusion

Dans ce chapitre, nous avons décrit la démarche expérimentale adoptée pour réaliser un système d'annotation automatique d'images. En effet, après avoir implémenté les algorithmes de la classification, de la description du contenu visuel des images et de la segmentation, nous avons réalisé plusieurs expériences qui nous ont conduit à définir l'architecture la plus performante en termes d'annotation d'images. Cette architecture originale est définie par la combinaison des descripteurs (forme, couleur et texture), combinaison des réseaux de neurones avec les réseaux bayésiens et le regroupement des régions adjacentes après la segmentation. Le système d'annotation automatique d'images, ainsi réalisé, est utilisé pour annoter des images contenant des objets appartenant aux trois bases de données images utilisées lors de l'apprentissage. Les images qui sont correctement segmentées sont bien annotées.

Chapitre VI CONCLUSION GENERALE ET PERSPECTIVES

« Méfie-toi des images. Ce n'est pas parce qu'on photographie le réel qu'on montre la réalité. »

[Sophie Bassignac]

Contenu du Chapitre

VI.1 Conclusion générale

Dans ce travail, nous nous sommes intéressés à la conception et la réalisation d'un système d'annotation automatique d'images permettant de réduire le fossé sémantique existant entre la description visuelle et la sémantique associée à une image. Ainsi, nous nous sommes penchés, au début de ce travail, sur l'étude bibliographique concernant l'annotation d'images. Cette étude nous a permis de concevoir la structure du système à réaliser. Pour rendre ce système opérationnel, nous avons implémenté plusieurs algorithmes d'extraction de paramètres descriptifs du contenu visuel des images, de classification et de segmentation. Ces algorithmes sont appliqués, dans un premier temps, à la reconnaissance des caractères Tifinagh. Cette application nous a permis de mettre en lumière le comportement de chaque descripteur et de chaque classificateur ainsi que de leur combinaison. En suite, afin de mettre en œuvre le système d'annotation, nous les avons utilisé pour réaliser plusieurs expériences d'annotation d'images de trois bases de données images ETH-80, COIL-100 et NATURE. L'analyse des résultats obtenus a mis en évidence la possibilité d'améliorer le taux d'annotation en combinant ou en fusionnant les descripteurs de forme, de texture et de couleur, et en combinant les réseaux de neurones avec les réseaux bayésiens. Pour explorer cette piste, nous avons commencé par réaliser des expériences d'annotation en combinant ou

fusionnant les descripteurs pour chacun des classificateurs mentionnés ci-dessus. L'étude des résultats émanant de ces expériences a montré que les deux procédures, à savoir la combinaison ou la fusion des descripteurs, permettent effectivement d'améliorer le taux d'annotation et que la combinaison surclasse la fusion. Ainsi, nous avons opté pour la combinaison des descripteurs pour réaliser d'autres expériences d'annotation en combinant les deux classificateurs considérés. Les résultats de ces expériences ont révélé une nette amélioration du taux d'annotation et que l'un des facteurs qui engendre les erreurs qui persistent, réside dans la segmentation. Pour remédier à ce problème, nous avons effectué le regroupement des régions de couleurs obtenues après la segmentation. L'application de cette technique a amélioré davantage le taux d'annotation.

Les études menées au cours de ce travail nous ont permis de réaliser un système doté d'une architecture modulaire et performante en termes d'annotation d'images. Cette architecture originale s'articule autour :

- de la combinaison du pouvoir discriminatif des réseaux de neurones avec le caractère génératif des réseaux bayésiens permettant ainsi de tirer bénéfice de leur complémentarité. La prise de la décision d'annotation est réalisée par le vote des différents classificateurs combinés.

- de la combinaison des descripteurs de forme (les moments de Legendre), de texture (matrice de Cooccurrence) et de couleur (les histogrammes de couleur RGB).

- du regroupement des régions de couleurs d'un objet après la segmentation par croissance des régions.

- de l'annotation d'un objet par sa couleur dominante afin d'enrichir l'interprétation sémantique d'une image.

Le système ainsi réalisé est un système modulaire, évolutif et non figé. Il permet l'intégration d'autres descripteurs du contenu visuel des images et d'autres méthodes de segmentation pour réduire davantage le fossé sémantique en particulier lorsqu'on passe à l'annotation de larges bases de données d'images.

Une évaluation du système proposé est réalisée à partir d'un corpus d'images différent de celui utilisé au cours de l'apprentissage. L'évaluation a montré que notre système permet d'obtenir des résultats probants.

En conclusion, la principale contribution de cette thèse réside dans la proposition d'une architecture originale permettant la réalisation d'un système d'annotation automatique d'images performant.

VI.2 Perspectives

A cours terme, nous envisageons passer à l'échelle, c'est-à-dire utiliser notre système pour annoter les images de larges bases. Dans ce cas, nous aurons une large variabilité visuelle intra-concept et une grande similarité visuelle inter-concept, qui conduisent souvent à des annotations imparfaites. Ainsi, pour maintenir une bonne performance en termes de taux d'annotation, nous comptons explorer les pistes d'amélioration suivantes :

- Intégrer d'autres descripteurs qui pourraient s'avérer utiles afin de perfectionner la description du contenu visuel des images.
- Améliorer la segmentation afin d'extraire correctement les objets présents dans une image complexe.
- Inclure une méthode de désambiguïsation basée sur une mesure de similarité sémantique qui intègre plusieurs sources d'information : visuelle, contextuelle et spatiale.

Après cette étape d'amélioration dudit système, nous allons :

- Le mettre en ligne sur le Web en utilisant la plateforme Java pour qu'il soit à la disposition d'une large communauté,
- L'utiliser pour développer un système de recherche d'image Visio-textuelle.

ANNEXES

Contenu des annexes

Annexe A CARACTERES TIFINAGH

La langue amazighe est représentée aujourd'hui par ses variantes parlées par les personnes appelées berbères ou amazighs. Ces variantes sont présentes au Maroc, l'Egypte, l'Algérie, la Tunisie, le Niger et le Mali. Le script berbère utilisé est appelé alphabets Tifinagh, et n'est pas tout à fait unie. L'écriture berbère ou amazighe connaît un certain nombre de variations temporelles et spatiales. Il y' a une trentaine de variétés. Au Maroc, la population berbérophone est divisée entre plusieurs dialectes: Tarifite dans le nord, le tamazight dans le Haut Atlas et le Moyen et le Tachelhite dans le centre du pays. Les Berbères marocains utilisent aussi les caractères Tifinagh comme motifs ornementaux dans l'artisanat (tapis, bijoux, poterie ...) et pour décorer l'intérieur des maisons traditionnelles. L'Institut royal de la culture amazighe[6] (IRCAM) a proposé la normalisation de l'alphabet Tifinagh à l'Organisation internationale de normalisation depuis 2004. L'alphabet Tifinagh adoptée par l'Institut royal de la culture amazighe (IRCAM) est composé de trente-trois caractères représentant les consonnes et les voyelles, comme indiqué dans la Figure A-1.

[6] http://www.ircam.ma/

Character number	Character	Character number	Character	Character number	Character
1	ⵔ	12	ⵍ	23	ⵕ
2	ⵒ	13	ⵎ	24	ⵖ
3	ⵗ	14	ⵏ	25	ⵘ
4	ⵗⵯ	15	ⵐ	26	ⵙ
5	ⵚ	16	ⵛ	27	ⵜ
6	ⵟ	17	ⵉ	28	ⵡ
7	ⵠ	18	ⵅ	29	ⵢ
8	ⵣ	19	ⵆ	30	ⵤ
9	ⵥ	20	ⵇ	31	ⵣ
10	ⵥⵯ	21	ⵈ	32	ⵊ
11	ⵧ	22	ⵓ	33	ⵋ

Figure A-1 : Caractères Tifinagh adoptées par l'Institut Royal de la Culture Amazighe (IRCAM) au Maroc.

Aujourd'hui, le travail de l'informatisation de la recherche sur la langue amazighe est concentré principalement au Maroc et en Algérie. Afin de développer et garder le patrimoine culturel de la population berbère, le Maroc dispose de trois centres de recherches dans ce domaine, à savoir : l'Institut royal de la culture amazighe (IRCAM) à Rabat, l'Université Sultan Moulay Slimane de Béni Mellal et l'Université Ibn Zohr à Agadir. Durant les dernières années, de nombreux chercheurs ont publié annuellement plusieurs publications de valeur dans le domaine de la reconnaissance optique de caractères Tifinagh (OCR).

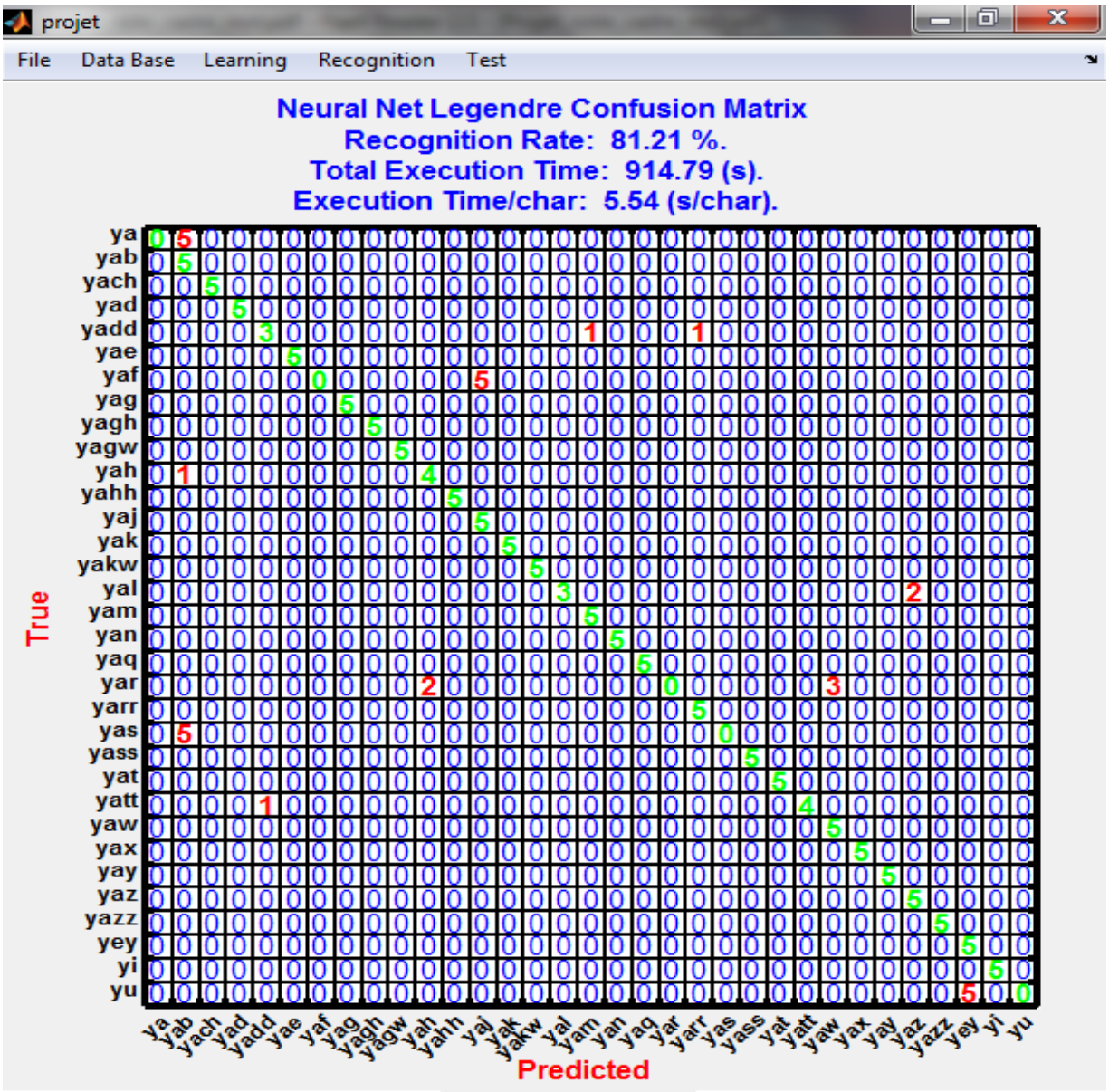


Figure A-2 : Matrice de confusion dans le cas d'utilisation des moments de Legendre avec les réseaux de neurones.

Pour se familiariser avec les différents descripteurs utilisé, dans cette thèse, pour l'annotation automatique d'images (les moments de Hu, de Zernike et de Legendre, la texture et les descripteurs GIST) ainsi que les différents classificateurs (Les réseaux de neurones, les réseaux bayésiens, les SVM et les k-plus proches voisins), ils sont appliqués dans un premier temps à la reconnaissance des caractères Tifinagh. De bons résultats ont été obtenus dans ce domaine. La Figure A-2 présente la matrice de confusion dans le cas d'utilisation des moments de Legendre avec les réseaux de neurones tandis que la Figure A-3 illustre la matrice de confusion dans le cas d'utilisation de la combinaison de plusieurs descripteurs et classificateurs.

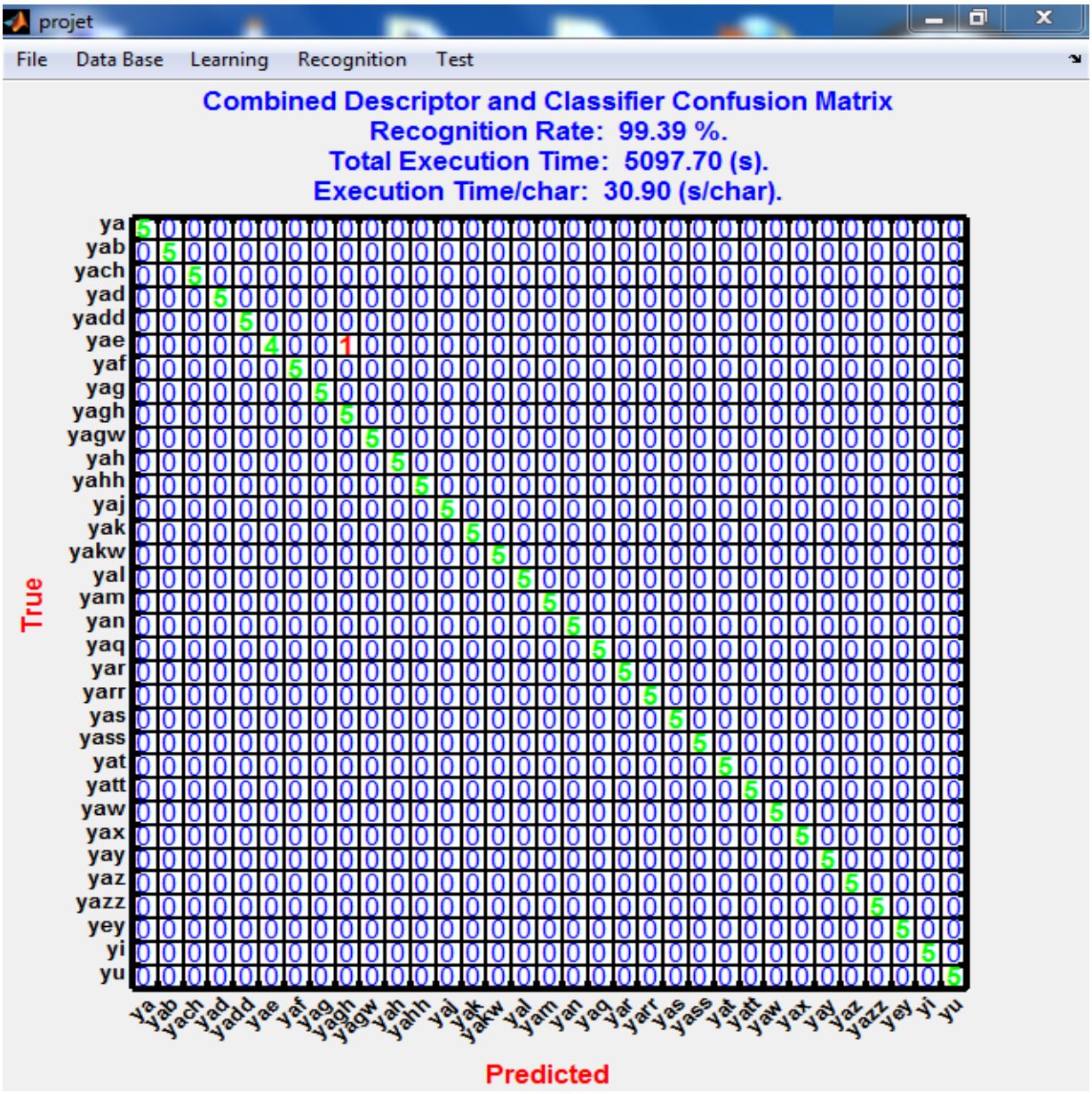


Figure A-3 : Matrice de confusion dans le cas d'utilisation de la combinaison de plusieurs descripteurs et classificateurs

À partir des deux figures, On peut voir une augmentation considérable du taux de reconnaissance. Le Tableau A-1 résume les résultats obtenus pour la reconnaissance des caractères Tifinagh en utilisant les différentes approches de description et de classification. Pour plus de détail, le lecteur peut se référer aux publications que nous avons réalisées [***117***], [***237***], [***238***], [***239***], [***240***], [***241***].

On remarque aussi que le temps d'exécution pour cetaines approches de description telles que les moments de legendre est prohibitif.

Tableau A-1 : Taux de reconnaissance, taux d'erreur et temps d'exécution approximatifs de chaque approche de description et chaque approche de classification des caractères Tifinagh.

	Approche de Description	**Approche de Classification**				
		Neural Network	***Nearest Neighbour***	***SVM One***	***SVM All***	***Bayesian Network***
Taux de reconnaissance (%)	Hu	72.73	83.64	56.97	41.82	83.64
	Zernike	80.61	95.76	69.70	87.88	95.76
	Legendre	81.21	97.58	49.70	78.79	94.55
	Walsh	68.48	69.09	65.45	42.42	69.09
	Texture	72.12	86.06	42.42	65.45	86.06
	GIST	30.03	98.18	36.36	83.64	98.18
	Approche basée sur la combinaison de 3 descripteurs et 4 classificateurs					98.79
	Approche basée sur la combinaison de 6 descripteurs et 5 classificateurs					99.39
	Approche basée sur la combinaison des descripteurs et classificateurs ayant un taux de reconnaissance superieur à 80%					100.00
Taux d'Erreur (%)	Hu	17.86	16.36	43.03	58.18	16.36
	Zernike	19.39	4.24	30.30	12.12	4.24
	Legendre	18.79	2.42	50.30	21.21	5.45
	Walsh	31.52	30.91	34.55	57.58	30.91
	Texture	27.88	13.94	57.58	34.55	13.94
	GIST	69.97	1.82	63.64	16.36	01.82
	Approche basée sur la combinaison de 3 descripteurs et 4 classificateurs					1.21
	Approche basée sur la combinaison de 6 descripteurs et 5 classificateurs					0.61
	Approche basée sur la combinaison des descripteurs et classificateurs ayant un taux de reconnaissance superieur à 80%					0.00
Temps d'Execution (s)	Hu	8.34	1.42	94.01	3.36	71.28
	Zernike	859.92	943.90	1152.36	754.65	645.70
	Legendre	853.16	805.33	892.41	806.00	1112.17
	Walsh	15.34	2.36	137.23	8.39	476.88
	Texture	14.60	3.43	113.42	9.10	528.65
	GIST	29.68	25.27	153.33	26.56	298.62
	Approche basée sur la combinaison de 3 descripteurs et 4 classificateurs					1900.17
	Approche basée sur la combinaison de 6 descripteurs et 5 classificateurs					3503.44
	Approche basée sur la combinaison des descripteurs et classificateurs ayant un taux de reconnaissance superieur à 80%					2513.59

Annexe B INDICES D'HARALICK

Haralick a introduit quatorze attributs de texture extraits des matrices de cooccurrences chromatiques. Pour chacune des deux composantes couleurs C_1 *et* C_2, ces attributs sont les suivants [***158***] :

B.1 Second moment angulaire

Le second moment angulaire (ou énergie) mesure l'homogénéité de l'image. Plus cette valeur est faible, moins l'image est uniforme et dans ce cas, il existe beaucoup de transitions de couleurs. Il est défini par :

$$f_1^{C,C'} = \sum_{i=0}^{T-1}\sum_{j=0}^{T-1}\left(M_{k,l}^{C,C'}([I],i,j)\right)^2 \qquad \textbf{(B.1)}$$

B.2 Contraste

Le contraste (ou inertie) mesure les variations locales des couleurs. Si ces variations sont importantes, alors le contraste sera élevé. Ce paramètre permet aussi de caractériser la dispersion des valeurs de la matrice de cooccurrences par rapport à sa diagonale principale. Il est donné par :

$$f_2^{C,C'} = \sum_{n=0}^{T-1} n^2 \left(\sum_{\substack{i=0 \\ |i-j|=n}}^{T-1}\sum_{j=0}^{T-1} M_{k,l}^{C,C'}([I],i,j) \right) \qquad \textbf{(B.2)}$$

B.3 Corrélation

Ce paramètre permet de déterminer si certaines colonnes de la matrice sont égales, c'est-à-dire s'il existe des dépendances linéaires dans l'image. En effet, plus les valeurs sont uniformément distribuées dans la matrice de cooccurrences chromatique et plus la corrélation est importante. Il est définit par :

$$f_3^{C,C'} = \frac{\sum_{i=0}^{T-1}\sum_{j=0}^{T-1}(i-\mu_x)(j-\mu_y)M_{k,l}^{C,C'}([I],i,j)}{\sigma_x\sigma_y} \quad \textbf{(B.3)}$$

Où σ_x *et* σ_y sont respectivement les écarts type de $M_x^{C,C'}([I],i)$ et $M_y^{C,C'}([I],j)$,

$$\mu_x = \frac{\sum_{i=0}^{T-1} i \cdot M_x^{C,C'}([I],i)}{\sum_{i=0}^{T-1} M_x^{C,C'}([I],i)}$$

$$\mu_y = \frac{\sum_{j=0}^{T-1} j \cdot M_y^{C,C'}([I],j)}{\sum_{j=0}^{T-1} M_y^{C,C'}([I],j)}$$

Avec

$$M_x^{C,C'}([I],i) = \sum_{j=0}^{T-1} M_{k,l}^{C,C'}([I],i,j)$$

Et $M_y^{C,C'}([I],j) = \sum_{i=0}^{T-1} M_{k,l}^{C,C'}([I],i,j)$

B.4 Variance

La variance mesure la répartition des couleurs autour de la valeur moyenne. Plus ce paramètre est élevé et plus importants sont les écarts entre les valeurs et la moyenne. Elle est définie par :

$$f_4^{C,C'} = \frac{1}{T^2}\sum_{i=0}^{T-1}\sum_{j=0}^{T-1}\left(M_{k,l}^{C,C'}([I],i,j)-\mu\right) \quad \textbf{(B.4)}$$

Où μ est la moyenne de tous les coefficients de la matrice $M_{k,l}^{C,C'}([I],i,j)$.

B.5 Moment différentiel inverse

Ce paramètre a un comportement inverse de celui du contraste. En effet, plus la texture possède de régions homogènes et plus le moment différentiel inverse est élevé. Il est définit par :

$$f_5^{C,C'} = \sum_{i=0}^{T-1}\sum_{j=0}^{T-1} \frac{1}{1+(i-j)^2} M_{k,l}^{C,C'}([I], i, j) \quad \textbf{(B.5)}$$

B.6 Moyenne des sommes

La moyenne des sommes est donnée par :

$$f_6^{C,C'} = \sum_{n=0}^{2(T-1)} n \left(\sum_{\substack{i=0 \\ i+j=n}}^{T-1}\sum_{j=0}^{T-1} M_{k,l}^{C,C'}([I], i, j) \right) \quad \textbf{(B.6)}$$

Ou bien :

$$f_6^{C,C'} = \sum_{n=0}^{2(T-1)} n \cdot M_{x+y}^{C,C'}([I], n) \quad \textbf{(B.7)}$$

Où on a :

$$M_{x+y}^{C,C'}([I], n) = \sum_{\substack{i=0 \\ i+j=n=0,1,\cdots,2(T-1)}}^{T-1}\sum_{j=0}^{T-1} M_{k,l}^{C,C'}([I], i, j)$$

B.7 Entropie des sommes

L'entropie des sommes est donnée par :

$$f_7^{C,C'} = -\sum_{n=0}^{2(T-1)} M_{x+y}^{C,C'}([I], n) \cdot \log\left\{M_{x+y}^{C,C'}([I], n)\right\} \quad \textbf{(B.8)}$$

Où on a :

$$M_{x+y}^{C,C'}([I], n) = \sum_{\substack{i=0 \\ i+j=n=0,1,\cdots,2(T-1)}}^{T-1}\sum_{j=0}^{T-1} M_{k,l}^{C,C'}([I], i, j)$$

B.8 Variance des sommes

La variance des sommes est donnée par :

$$f_8^{C,C'} = \sum_{n=0}^{2(T-1)} \left(1 - f_7^{C,C'}\right)^2 \cdot M_{x+y}^{C,C'}([I], n) \quad \textbf{(B.9)}$$

Où on a :

$$M_{x+y}^{C,C'}([I],n)=\sum_{\substack{i=0\\ i+j=n=0,1,\cdots,2(T-1)}}^{T-1}\sum_{j=0}^{T-1} M_{k,l}^{C,C'}([I],i,j)$$

B.9 Entropie

L'entropie mesure la complexité de l'image. Lorsque les valeurs de la matrice de cooccurrences sont presque toutes égales, l'entropie est élevée. Elle permet ainsi de caractériser le degré de granulation de l'image. En effet, plus l'entropie est grande et plus la granulation est grossière. Elle est définie par :

$$f_9^{C,C'}=-\sum_{i=0}^{T-1}\sum_{j=0}^{T-1} M_{k,l}^{C,C'}([I],i,j)\cdot\log\left\{M_{k,l}^{C,C'}([I],i,j)\right\} \qquad \textbf{(B.10)}$$

B.10 Entropie des différences

L'entropie des différences est donnée par :

$$f_{10}^{C,C'}=-\sum_{n=0}^{2(T-1)} M_{x-y}^{C,C'}([I],n)\cdot\log\left\{M_{x-y}^{C,C'}([I],n)\right\} \qquad \textbf{(B.11)}$$

Avec :

$$M_{x-y}^{C,C'}([I],n)=\sum_{\substack{i=0\\ |i-j|=n=0,1,\cdots,(T-1)}}^{T-1}\sum_{j=0}^{T-1} M_{k,l}^{C,C'}([I],i,j)$$

B.11 Variance des différences

La variance des différences est donnée par :

$$f_{11}^{C,C'}=\sum_{n=0}^{T-1}\left(n-f_{10}^{C,C'}\right)^2\cdot M_{x-y}^{C,C'}([I],n) \qquad \textbf{(B.12)}$$

Avec :

$$M_{x-y}^{C,C'}([I],n)=\sum_{\substack{i=0\\ |i-j|=n=0,1,\cdots,(T-1)}}^{T-1}\sum_{j=0}^{T-1} M_{k,l}^{C,C'}([I],i,j)$$

B.12 Information sur la corrélation 1

L'indice de l'information sur la corrélation 1 est donné par :

$$f_{12}^{C,C'} = \frac{f_9^{C,C'} - HXY1}{\max\{HX, HY\}} \quad \textbf{(B.13)}$$

Avec :

$$HXY1 = -\sum_{i=0}^{T-1}\sum_{j=0}^{T-1} M_{k,l}^{C,C'}([I], i, j) \cdot \log\left\{M_x^{C,C'}([I], i)\, M_y^{C,C'}([I], j)\right\}$$

$$HX = -\sum_{i=0}^{T-1}\sum_{j=0}^{T-1} M_x^{C,C'}([I], i) \cdot \log\left\{M_x^{C,C'}([I], i)\right\}$$

$$HY = -\sum_{i=0}^{T-1}\sum_{j=0}^{T-1} M_y^{C,C'}([I], j) \cdot \log\left\{M_y^{C,C'}([I], j)\right\}$$

B.13 Information sur la corrélation 2

L'indice de l'information sur la corrélation 2 est donné par :

$$f_{13}^{C,C'} = \left(1 - \exp\left[-2\left(HXY2 - f_9^{C,C'}\right)\right]\right)^{\frac{1}{2}} \quad \textbf{(B.14)}$$

Avec :

$$HXY2 = -\sum_{i=0}^{T-1}\sum_{j=0}^{T-1} M_x^{C,C'}([I], i)\, M_y^{C,C'}([I], j) \cdot \log\left\{M_x^{C,C'}([I], i)\, M_y^{C,C'}([I], j)\right\}$$

B.14 Coefficient de corrélation maximal

Le coefficient de corrélation maximal est donné par :

$$f_{14}^{C,C'} = \left(1^{ère}\ plus\ grande\ valeur\ propre\ de\ Q\right)^{\frac{1}{2}} \quad \textbf{(B.15)}$$

Avec :

$$Q(i, j) = -\sum_{n=0}^{T-1} \frac{M_{k,l}^{C,C'}([I], i, n)\, M_{k,l}^{C,C'}([I], j, n)}{M_x^{C,C'}([I], i)\, M_y^{C,C'}([I], n)}$$

Les attributs $f_6^{C,C'}$, $f_7^{C,C'}$, $f_8^{C,C'}$, $f_{10}^{C,C'}$, $f_{11}^{C,C'}$, $f_{12}^{C,C'}$, $f_{13}^{C,C'}$ et $f_{14}^{C,C'}$ apportent des informations supplémentaires sur les degrés d'homogénéité et de complexité de l'image, ainsi que sur la corrélation.

BIBLIOGRAPHIE

[1] Michael G Krauze. Intelletual problems of indexing picture collections. Audiovisual Librarian, 14(4), pp. 73–81, 1998.

[2] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content based image retrieval at the end of the early years," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 12, pp. 1349–1380, 2000.

[3] N. Balasubramanian, A.R. Diekema, and A.A. Goodrum. Analysis of user image descriptions and automatic image indexing vocabularies: An exploratory study. In International Workshop on Multidisciplinary Image, Video, and Audio Retrieval and Mining. Sherbrooke, Quebec, Canada, October 25-26, 2004.

[4] Abebe Rorissa. User-generated descriptions of ind ividual images versus labels of groups of images : A comparison using basic level theory. Information Processing & Management, 44(5) :1741–1753, 2008.

[5] Anthony Ventresque. Une mesure de similarité sémantique utilisant des résultats de psychologie. In CORIA, Session Jeunes Chercheurs, pages 371–376, 2006.

[6] P. O. Ogunbona, "Visual information processing and content management: an overview," International Journal for Information and Systems Sciences, vol. 3, no. 3, pp. 349–364, 2007.

[7] F. Long, H. Zhang, and D. D. Feng, "Fundamentals of content-based image retrieval," in Multimedia Information Retrieval and Management – Technological Fundamentals and Applications, Berlin: Springer, pp. 95-120, 2003.

[8] J. P. Eakins, "Retrieval of still images by content," in Lecture Notes in Computer Science: Lectures on Information Retrieval, vol. 1980/2001, pp. 111–138, Berlin: Springer, 2001.

[9] Lei Wang, Li Liu and Latifur Khan. Automatic image annotation and retrieval using subspace clustering algorithm. In MMDB'04: Proceedings of the 2nd ACM international workshop on multimedia databases, pp. 100-108, New York, USA, 2004.

[10] Masashi Inoue. On the need for annotation-based image retrieval. In Workshop on Information Retrieval in Context (IRiX) Sheffield, UK, pp.44-46, July 2004.

[11] Marin Ferecatu, Nozha Boujemaa, and Michel Crucianu. Hybrid visual and conceptual image representation within active relevance feedback context. In 7th ACM SIGMM International Workshop on Multimedia Information Retrieval (MIR'05), Singapore, pp.209-216, 2005.

[12] Kerry Rodden and Kenneth R. Wood. How do people manage t heir digital photographs ? In ACM CHI '03 : Proceedings of the SIGCHI conference on Human factors in computing systems, pages 409–416, New York, NY, USA, 2003. ACM.

[13] Roelof van Zwol. Flickr : Who is looking ? In IEEE/WIC/ACM International Conference on Web Intelligence, Pages 184-190, IEEE Computer Society Washington, DC, USA, 2007.

[14] Radu Andrei Negoescu and Daniel Gatica-Perez. Analyzing flickr groups. In CIVR '08: Proceedings of the 2008 international conference on Content-based image and video retrieval, pages 417–426, New York, NY, USA, 2008. ACM.

[15] Robert Ortgies, Christoph Dosch, Jan Nesvadba, Adolf Proidl, Henri Gouraud, Pieter van der Linden, Nozha Boujemaa, Jussi Karlgren, Ramon Compano, Joachim Köhler, Paul King, and David Lowen. Chorus d3.3 - vision document, intermediate, results of the 3rd think-tank, pages 6–22, Novembre 2008.

[16] G. Ciocca, C. Cusano, and R. Schettini. Semantic classification, low level features and relevance feedback for content-based image retrieval. In Multimedia Content Access : Algorithms and Systems III, IS&T/SPIE Symposium on Electronic Imaging, Volume 7255, pp. 72550D, San Jose, 2009.

[17] Joao Magalhaes, Fabio Ciravegna, and Stefan Ruger. Exploring multimedia in a keyword space. In MM '08 : Proceeding of the 16th ACM international conference on Multimedia, pages 101–110, New York, NY, USA, 2008. ACM.

[18] N. Boujemaa and M. Ferecatu. Evaluation des systèmes de traitement de l'information, chapitre : Evaluation des systèmes de recherche par le contenu visuel : pertinence et critères. ISBN 2-7462-0862-8. Hermes Sciences, 2004.

[19] Sabine Barrat, Modèles graphiques probabilistes pour la reconnaissance de formes, thèse de l'université Nancy 2, Spécialité informatique, décembre 2009.

[20] Bryan C. Russell, Antonio Torralba, Kevin P. Murphy, and William T. Freeman. Labelme: A database and web-based tool for image annotation. Int. J. Comput. Vision, 77(1-3):157–173, May 2008.

[21] Luis Von Ahn & Laura Dabbish. Labeling images with a computer game. In CHI' 04, pages 319–326, New York, NY, USA, 2004.

[22] Luis Von Ahn. Games with a Purpose. Computer, vol. 39, no. 6, pages 92–94, 2006.

[23] J. P. Fan, Y. Gao & H. Z. Luo. Integrating Concept Ontology and Multitask Learning to Achieve More Effective Classifier Training for Multilevel Image Annotation. IEEE Trans. Image Processing, vol. 17, no. 3, pages 407–426, 2008.

[24] X. J. Wang, L. Zhang, X. Li & W. Y. Ma. Annotating Images by Mining Image Search Results. IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 11, pages 1919–1932, 2008.

[25] R. C. F. Wong & C. H. C. Leung. Automatic Semantic Annotation of Real-World Web Images. IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 11, pages 1933–1944, 2008.

[26] J.Z Wang, Jia Li. Real-Time Computerized Annotation of Pictures, Pattern Analysis and Machine Intelligence, IEEE Transactions on Volume: 30, Issue: 6, pp. 985-1002, 2008

[27] Ballan, Lamberto; Bertini, Marco; Uricchio, Tiberio; Del Bimbo, Alberto, Social media annotation, 11th International Workshop on Content-Based Multimedia Indexing 2013 (CBMI), pp. 229-235, 17-19 June 2013.

[28] Qi Mao; Tsang, I.W.-H.; Shenghua Gao, "Objective-Guided Image Annotation," Image Processing, IEEE Transactions on , vol.22, no.4, pp.1585-1597, April 2013.

[29] Kobus Barnard, Pinard Duygulu, David Forsyth, Nando de Freitas, David M. Blei and Michael I. Jordan. Matching words and pictures. Journal of machine learning, Vol. 3, pp. 1107-1135, 2003.

[30] J. Jeon, V. Lavrenko and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In SIGIR'03 : Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 119-126, New York, USA, 2003.

[31] Florent Monay, Daniel Gatica-Perez. On image auto-annotation with latent space models. In multimedia'03 : Proceedings of 11th ACM International conference on multimedia, pp. 275-278, New York, USA, 2003.

[32] E. Chang, Kingshy Goh, G. Sychay, Gang Wu. CBSA : Content based soft annotation for multimodal image retrieval using Bayes point machines. Circuits and systems for video technology, IEEE Transactions on, Vol. 13, no. 1, pp. 26-38, 2003.

[33] S. L. Feng, R. Manmatha and V. Lavrenko. Multiple Bernoulli relevance models for image and video annotation. In proceeding of the 2004 IEEE computer society conference on computer vision and pattern recognition, CVPR'04, pp. 1002-1009, USA, 2004.

[34] Lavrenko, V., Manmatha, R., and Jeon, J. A model for learning the semantics of pictures. In Neural Information Processing Systems (NIPS'03). MIT Press. 2003.

[35] G. Carneiro, A. B. Chan, P. J. Moreno, and N. Vasconcelos, Supervised learning of semantic classes for image annotation and retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence. vol. 29, no. 3, pp. 394–410, 2007.

[36] Zhang, D., Islam, M. M., and Lu, G., A review on automatic image annotation techniques. Pattern Recognition, 45(1): pp. 346 – 362, 2012.

[37] M. Szummer and R. W. Picard, "Indoor-outdoor image classification," in Proc. of IEEE International Workshop on Content-Based Access of Image and Video Database, pp. 42–51, 1998.

[38] A. Vailaya and et.al., "Image classification for content-based indexing," IEEE Trans. Image Process., vol. 10, no. 1, pp. 117–130, 2001.

[39] A. Bosch, A. Zisserman, and X. Muoz, "Scene classification using a hybrid generative/discriminative approach," IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no. 4, pp. 712–727, Apr. 2008.

[40] F.-F. Li and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 524–531, 2005.

[41] L.-J. Li, R. Socher, and F.-F. Li, "Towards total scene understanding: classification, annotation, segmentation in an automatic framework," in Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Florida, USA, pp. 2036–2043, Jun. 2009.

[42] Florent Monay, Daniel Gatica-Perez, "pLSA-based image auto-annotation: constraining the latent space," in Proceedings of the ACM International Conference on Multimedia, pp. 348–351, 2004.

[43] A. Torralba, "Contextual priming for object detection," International Journal of Computer Vision, vol. 53, no. 2, pp. 169–191, 2003.

[44] P. Duygulu, K. Barnard, J. F. G. de Freitas, D. A. Forsyth et al., "Object recognition as machine translation: learning a lexicon for a fixed image vocabulary," in Proceedings of the European Conference on Computer Vision (ECCV), pp. 97–112, 2002.

[45] P. Carbonetto, N. Freitas, and K. Barnard, "A statistical model for general contextual object recognition," in Proceedings of the European Conference on Computer Vision (ECCV), pp. 350–362, 2004.

[46] D. M. Blei and M. I. Jordan, "Modeling annotated data," in Proceedings of the ACM International Conference on Research and Development in Informaion Retrieval (SIGIR), pp. 127–134, 2003.

[47] P. Viola, and M. Jones, Rapid object detection using a boosted cascade of simple features. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2001), volume 1, pages I–511 –I–518, 2001.

[48] A. Mohan, C. Papageorgiou, and T. Poggio, Example based object detection in images by components. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI'01), 23(4): pp. 349 –361, 2001.

[49] N. Dalal, and B. Triggs, Histograms of oriented gradients for human detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2005), volume 1, pages 886 –893, 2005.

[50] A. Torralba, K. Murphy, and W. Freeman, Sharing visual features for multiclass and multiview object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI'07), 29(5): pp. 854 –869, 2007.

[51] Y. Wei, and L. Tao, Efficient histogram-based sliding window. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10), pages 3003–3010, 2010.

[52] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, Object detection with discriminatively trained part-based models. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI'10), 32(9): pp. 1627 –1645, 2010.

[53] C. Galleguillos, and S. Belongie, Context based object categorization: A critical survey. Computer Vision and Image Understanding, 114(6): pp. 712–722, 2010.

[54] G. Carneiro, and N. Vasconcelos, Formulating semantic image annotation as a supervised learning problem. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 2, pages 163–168 vol. 2, 2005.

[55] G. Griffin, and P. Perona, Learning and using taxonomies for fast visual categorization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'08), pp. 1–8, 2008.

[56] Q. Zhang, S. A. Goldman, W. Yu, and J. Fritts, Content-based image retrieval using multiple-instance learning. In Proceedings of the Nineteenth International Conference on Machine Learning (ICML'02) , pages 682–689, 2002.

[57] Y. Chen, and J. Z. Wang, Image categorization by learning and reasoning with regions. Journal of Machine Learning Research, 5: pp. 913–939, 2004.

[58] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning. Springer-Verlag, 2001.

[59] J. Li, and J. Wang, Automatic linguistic indexing of pictures by a statistical modeling approach. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 25(9): pp. 1075 – 1088, 2003.

[60] J.Tang, R. Hong, S. Yan, T.-S. Chua, G.-J. Qi, and R. Jain, Image annotation by knn-sparse graph-based label propagation over noisily tagged web images. ACM Transactions on Intelligent Systems and Technology, 2(2): pp. 14:1–14:15, 2011.

[61] L. B. Romdhane, H. Bannour, and B. el Ayeb, IMIOL: a system for indexing images by their semantic content based on possibilistic fuzzy clustering and adaptive resonance theory neural networks learning. Applied Artificial Intelligence, 24(9): pp. 821–846, 2010.

[62] K. Barnard, and D. A. Forsyth, Learning the semantics of words and pictures. In Proceedings of the International Conference on Computer Vision (ICCV'11), pages 408–415, 2001.

[63] Y. Feng, and M. Lapata, Topic models for image annotation and text illustration. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, HLT'10, pages 831–839, 2010.

[64] F. Monay, and D. Gatica-Perez, PLSA based image auto-annotation: constraining the latent space. In Proceedings of the 12th annual ACM international conference on Multimedia (ACM MM'04) , pages 348–351, 2004.

[65] M. Guillaumin, J. Verbeek, and C. Schmid, Multimodal semi-supervised learning for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10), pages 902–909, 2010.

[66] H. Kück, P. Carbonetto, and N. de Freitas, A constrained semi-supervised learning approach to data association. In Proceedings of the European Conference on Computer Vision (ECCV'10) , pages 1–12, 2004.

[67] R. Fergus, Y. Weiss, and A. Torralba, Semi-supervised learning in gigantic image collections. In Neural Information Processing Systems (NIPS'09), pages 522–530, 2009.

[68] M. Wang, X.-S. Hua, Y. Song, X. Yuan, S. Li, and H.-J. Zhang, Automatic video annotation by semi-supervised learning with kernel density estimation. In Proceedings of the 14th annual ACM international conference on Multimedia (MM'06), pages 967–976, 2006.

[69] Z.-H. Zhou, D.-C. Zhan, and Q. Yang, Semi-supervised learning with very few labeled training examples. In Proceedings of the 22nd national conference on Artificial intelligence (AAAI'07), pages 675–680, 2007.

[70] X. Zhu, Semi-supervised learning literature survey. Technical report, University of Wisconsin-Madison, Department of Computer Science, 2006.

[71] J. Deng, A. C. Berg, K. Li, and L. Fei-Fei, What does classifying more than 10,000 image categories tell us? In Proceedings of the 11th European Conference on Computer Vision (ECCV'10), Part V, Pages 71-84, Springer-Verlag Berlin, 2010.

[72] M. Marszalek, and C. Schmid, Constructing category hierarchies for visual recognition. In Proceedings of the European Conference on Computer Vision (ECCV), pages 479–491, 2008.

[73] H. Cevikalp, New clustering algorithms for the support vector machine based hierarchical classification. Pattern Recognition Letters, 31(11): pp. 1285 – 1291, 2010.

[74] T. Gao, and D. Koller, Discriminative learning of relaxed hierarchy for large-scale visual recognition. In Proceedings of the International Conference on Computer Vision (ICCV'11), Pages 2072-2079, IEEE Computer Society Washington, DC, USA ©2011.

[75] M. Marszalek, and C. Schmid, Semantic hierarchies for visual object recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'07), pages 1 –7, 2007.

[76] L.-J. Li, C. Wang, Y. Lim, D. M. Blei, and F.-F. Li, Building and using a semantivisual image hierarchy. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10), pages 3336 –3343, 2010.

[77] S. Bengio, J. Weston, and D. Grangier, Label embedding trees for large multi-class tasks. In Advances in Neural Information Processing Systems (NIPS'10), pages 163–171, 2010.

[78] J. Deng, S. Satheesh, A. C. Berg, and F.-F. Li, Fast and balanced: Efficient label tree learning for large scale object recognition. In Advances in Neural Information Processing Systems (NIPS'11), pages 567–575, 2011.

[79] A. Zweig, and D. Weinshall, Exploiting object hierarchy: Combining models from different category levels. In Proceedings of the International Conference on Computer Vision (ICCV'07), pages 1 –8, 2007.

[80] J. Fan, Y. Gao, and H. Luo, Hierarchical classification for automatic image annotation. In international ACM SIGIR conference on Research and development in information retrieval (SIGIR'07), pages 111–118, 2007.

[81] A. Torralba, R. Fergus, and W. Freeman, 80 million tiny images: A large data set for nonparametric object and scene recognition. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 30(11): pp. 1958–1970, 2008.

[82] N. Maillot, and M. Thonnat, Ontology based complex object recognition. Image and Vision Computing, 26(1):102 – 113, 2008.

[83] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, ImageNet: a large-scale hierarchical image database. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'09), 2009.

[84] N. Gronau, M. Neta, and M. Bar, Integrated contextual representation for objects' identities and their locations. Journal of Cognitive Neuroscience, 20(3): pp. 371–388, 2008.

[85] C. Hudelot, J. Atif, and I. Bloch, Fuzzy spatial relation ontology for image interpretation. Fuzzy Sets and Systems, 159: pp. 1929–1951, 2008.

[86] B. Neumann, and R. Möller, On scene interpretation with description logics. Image Vision Computing, 26(1): pp. 82–101, 2008.

[87] A. Oliva, and A. Torralba, The role of context in object recognition. Trends in cognitive sciences, 11(12): pp. 520–527, 2007.

[88] S. Tollari, Indexation et recherche d'images par fusion d'informations textuelles et visuelles (Image indexing and retrieval by combining textual and visual informations). PhD thesis, Université du Sud Toulon-Var, 2006.

[89] Clinchant, S., Ah-Pine, J., and Csurka, G. (2011). Semantic combination of textual and visual information in multimedia retrieval. In Proceedings of the 1st ACM International Conference on Multimedia Retrieval , ICMR '11, pages 44:1–44:8, New York, NY, USA. ACM, 2011.

[90] Rohrbach, M., Stark, M., Szarvas, G., Gurevych, I., and Schiele, B. (2010). What helps? where? and why? semantic relatedness for knowledge transfer. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 910 –917, 2010.

[91] Znaidia, A., Shabou, A., Popescu, A., le Borgne, H., and Hudelot, C. (2012). Multimodal feature generation framework for semantic image classification. In Proceedings of the 2nd ACM International Conference on Multimedia Retrieval, ICMR '12, pages 38:1–38:8, New York, NY, USA. ACM, 2012.

[92] Tommasi, T., Orabona, F., and Caputo, B. (2008). Discriminative cue integration for medical image annotation. Pattern Recognition Letters, 29(15):1996 – 2002, 2008.

[93] Clinchant, S., Ah-Pine, J., and Csurka, G. (2011). Semantic combination of textual and visual information in multimedia retrieval. In Proceedings of the 1st ACM International Conference on Multimedia Retrieval , ICMR '11, pages 44:1–44:8, New York, NY, USA. ACM, 2011.

[94] N. Rasiwasia, J. Costa Pereira, E. Coviello, G. Doyle, G. R. Lanckriet, R. Levy, and N. Vasconcelos, A new approach to cross-modal multimedia retrieval. In Proceedings of the international conference on Multimedia (MM'10), pages 251–260. ACM, 2010.

[95] H. J. Escalante, C. A. Hérnadez, L. E. Sucar, and M. Montes, Late fusion of heterogeneous methods for multimedia image retrieval. In Proceedings of the 1st ACM international conference on Multimedia information retrieval (MIR'08), pages 172–179. ACM, 2008.

[96] M. J. Choi, J. Lim, A. Torralba, and A. Willsky, Exploiting hierarchical context on a large database of object categories. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10), pages 129 –136, 2010.

[97] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. Berg, and T. Berg, Baby talk: Understanding and generating simple image descriptions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 1601 –1608, 2011.

[98] E. Bruno, N. Moenne-Loccoz, and S. Marchand-Maillet, Design of multimodal dissimilarity spaces for retrieval of video documents. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI'08), 30(9): pp. 1520–1533, 2008.

[99] J. Ah-Pine, M. Bressan, S. Clinchant, G. Csurka, Y. Hoppenot, and J.-M. Renders, Crossing textual and visual content in different application scenarios. Multimedia Tools and Applications, 42: pp. 31–56, 2009.

[100] I. Horrocks, P. F. Patel-Schneider, and F. van Harmelen, From SHIQ and RDF to OWL: the making of a Web ontology language. Web Semantics: Science, Services and Agents on the World Wide Web, 1(1): pp. 7–26, 2003.

[101] F. Baader, What's new in description logics. Informatik Spektrum, 34: pp. 434–442, 2011.

[102] N. Simou, T. Athanasiadis, G. Stoilos, and S. Kollias, Image indexing and retrieval using expressive fuzzy description logics. Signal, Image and Video Processing, 2(4):321–335, 2008.

[103] S. Dasiopoulou, I. Kompatsiaris, and M. Strintzis, Applying fuzzy DLs in the extraction of image semantics. In S. Spaccapietra, and L. Delcambre, editors, Journal on Data Semantics XIV, volume 5880 of Lecture Notes in Computer Science, pages 105–132. Springer Berlin / Heidelberg, 2009.

[104] C. Hudelot, J. Atif, and I. Bloch, Integrating bipolar fuzzy mathematical morphology in description logics for spatial reasoning. In European Conference on Artificial Intelligence (ECAI'10), pages 497–502, 2010.

[105] A., Hamadi, G. Quenot, and P. Mulhem, Two-layers re-ranking approach based on contextual information for visual concepts detection in videos. In Content-Based Multimedia Indexing (CBMI), 2012 10th International Workshop on, pages 1–6. 2012.

[106] J. Martinet, Y. Chiaramella, and P. Mulhem, A relational vector space model using an advanced weighting scheme for image retrieval. Information Processing & Management, 47(3): pp. 391 – 414, 2011.

[107] L. Wu, X.-S. Hua, N. Yu, W.-Y. Ma, and S. Li, Flickr distance: A relationship measure for visual concepts. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 34(5):863 –875, 2012.

[108] J. Atif, C. Hudelot, and I. Bloch, Explanatory reasoning for image understanding using formal concept analysis and description logics. IEEE Transactions on Systems, Man and Cybernetics, pp. 1–19, 2013.

[109] A.-M. Tousch, S. Herbin, and J.-Y. Audibert, Semantic hierarchies for image annotation: A survey. Pattern Recognition, 45(1): pp. 333–345, 2012.

[110] Changbo Yang, Ming Dong and Farshad Fotouhi, Semantic feedback for interactive image retrieval, In MILTUMEDIA 05 : proceedings of the 13th annual ACM international conference on multimedia, pp. 415-418, New York, USA, 2005.

[111] Liu Wenyin, Susan Dumais, Yanfeng Sun, Hongjiang Zhang, Mary Czerwinski and Brent Field, Semi-Automatic Image Annotation, In INTERACT2001, 8th IFIP TC.13, Conference on Human-Computer Interaction, pp. 326-333, 2001.

[112] Ryszard S. Chora´s, Image Feature Extraction Techniques and Their Applications for CBIR and Biometrics Systems, International Journal Of Biology And Biomedical Engineering, Issue 1, Vol. 1, pp. 6-16, 2007.

[113] M. Oujaoura, B. Minaoui, M. Fakir, "A semantic approach for automatic image annotation," Intelligent Systems: Theories and Applications (SITA), 2013 8th International Conference on , vol., no., pp.1–8, 8-9 May 2013, doi: 10.1109/SITA.2013.6560800.

[114] Mustapha OUJAOURA, Brahim MINAOUI and Mohammed FAKIR, Multilayer Neural Networks and Nearest Neighbor Classifier Performances for Image Annotation, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 3, No. 11, pp.165–171, 2012. Published by The Science and Information Organization, New York, USA.

[115] Mustapha OUJAOURA, Brahim MINAOUI and Mohammed FAKIR. Article: Image Annotation using Moments and Multilayer Neural Networks. IJCA Special Issue on

Software Engineering, Databases and Expert Systems SEDEX(1):46-55, September 2012. Published by Foundation of Computer Science, New York, USA.

[116] Zijun Yang and C.-C. Jay Kuo, Survey on Image Content Analysis, Indexing, and Retrieval Techniques and Status Report of MPEG-7, Tamkang Journal of Science and Engineering, Vol. 2, No. 3, pp. 101-118, 1999.

[117] H. MOUDNI, M. ER-ROUIDI, Mustapha OUJAOURA and O. BENCHAREF, Recognition of Amazigh characters using SURF & GIST descriptors, (IJACSA) International Journal of Advanced Computer Science and Applications, special issue on selected papers from third international symposium on automatic amazigh processing SITACAM13, Vol. 3, No. 2, pp.41–44, 2013. Published by The Science and Information Organization, New York, USA.

[118] Bencharef, O.; Fakir, M.; Minaoui, B.; Hajraoui, A.; Oujaoura, M., "Color objects recognition system based on artificial neural network with Zernike, Hu & Geodesic descriptors," Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2012 6th International Conference on , vol., no., pp.338–343, 21-24 March 2012, doi: 10.1109/SETIT.2012.6481938.

[119] Yue Cao, Xiabi Liu, Jie Bing and Li Song, Using Neural Network to Combine Measures of Word Semantic Similarity for Image Annotation, IEEE International Conference on Information and Automation (ICIA), pp. 833 – 837, 2011.

[120] A. F. Smeaton, P. Over & W. Kraaij. High-Level Feature Detection from Video in TRECVid : a 5-Year Retrospective of Achievements. In Multimedia Content Analysis, Theory and Applications, pages 151–174. Springer Verlag, 2009.

[121] M. Everingham, J. Sivic & A. Zisserman. Taking the bite out of automated naming of characters in TV video. Image Vision Comput., vol. 27, no. 5, pages 545–559, 2009.

[122] N. Vandenbroucke. Segmentation d'images couleur par classification de pixels dans des espaces d'attributs colorimétriques adaptés. Application à l'analyse d'images de football. Thèse de doctorat, Université des sciences et technologies de Lille1, Décembre 2000.

[123] R. Joubert Olivier, Catégorisation Rapide Des Scènes Naturelles: L'objet, Le Contexte, Et Leurs Interactions, Thèse de doctorat de l'Université Toulouse III - Paul Sabatier, 30 Septembre 2008.

[124] S. Ouatara, Stratégie de segmentation d'images multi-composantes par analyse d'histogrammes multidimensionnels, Thèse de doctorat, Ecole doctorale d'Angers STIM, France, 2009.

[125] W. Zucker Steven. Region growing : Childhood and adolescence. Computer Graphics and Image Processing, vol. 5, no. 3, pages 382–399, 1976.

[126] Y. Shih Frank, Shouxian Cheng, Automatic seeded region growing for color image segmentation, Image and Vision Computing 23, pp. 877–886, 2005.

[127] J. McQueen. Some methods for classification and analysis of multivariate observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1:281-297, 1967.

[128] E. Forgy. Cluster analysis of multivariate data: Efficiency vs. interpretability of classifications. Biometrics, page 21:768-769, 1965.

[129] Rong Zhang and Alexander I. Rudnicky. A large scale clustering scheme for kernel k-means. In ICPR (4), pages 289-292, 2002.

[130] Aristidis Likas, Nikos A. Vlassis, and Jakob J. Verbeek. The global k-means clustering algorithm. Pattern Recognition, 36(2): pp. 451-461, 2003.

[131] B. Zhang and M. Hsu. K-harmonic means - a data clustering algorithm, pages 1-25, 1999.

[132] Greg Hamerly and Charles Elkan. Learning the k in k-means. In NIPS, pages 281-288, December 2003.

[133] Dan Pelleg and Andrew Moore. X-means: Extending K-means with efficient estimation of the number of clusters. In Proc. 17th International Conf. on Machine Learning, pages 727-734. Morgan Kaufmann, San Francisco, CA, 2000.

[134] Lior Rokach, Oded Maimon. Data Mining and Knowledge Discovery Handbook, Chapter 15: Clustering Methods. pp 321-352, Springer series, 2nd Edition, New York, October 1, 2010.

[135] Léon Bottou and Yoshua Bengio. Convergence properties of the k-means algorithms. In NIPS, pages 585-592, 1994.

[136] A. Tarsitano, Mahalanobis metrics for k-means algorithms. Convegno intermedio SIS, pages 9-11, 2003.

[137] D. Art, R. Gnanadesikan, and J. R Kettenring, Data-based metrics for hierarchical cluster analysis. Utilitas Mathematica, (21A):75-99, 1982.

[138] P.S. Bradley and Usama M. Fayyad. Refining initial points for k-means clustering. ICML '98 Proceedings of the Fifteenth International Conference on Machine Learning, Pages 91-99, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA 1998.

[139] Siddheswar Ray and Rose H. Turi. Determination of number of clusters in k-means clustering and application in colour image segmentation, in 4th International Conference on Advances in Pattern Recognition and Digital Techniques (ICAPRDT'99), 1999.

[140] R. Sinan Tumen, M. Emre Acer and T. Metin Sezgin, Feature Extraction and Classifier Combination for Image-based Sketch Recognition, Eurographics Symposium on Sketch-Based Interfaces and Modeling, pp. 1–8, 2010.

[141] F. L. Alt, Digital Pattern Recognition by Moments, J. Assoc. Computing Machinery, Vol. 9, pp. 240-258, 1962.

[142] M. K. Hu, Visual Problem Recognition by Moment Invariant, IRE.Trans. Inform. Theory, Vol. IT-8, pp. 179-187, Feb. 1962.

[143] Sun-Kyoo Hwang, Whoi-Yul Kim, A novel approach to the fast computation of Zernike moments, Pattern Recognition 39, pp. 2065 – 2076, 2006.

[144] A. Prata, W.V.T. Rusche, Algorithm for computation of Zernike polynomials expansion coefficients, Appl. Opt. 28, pp. 749–754, 1989.

[145] E.C. Kintner, On the mathematical properties of the Zernike polynomials, Opt. Acta. 23 (8), pp. 679–680, 1976.

[146] C.W. Chong, P. Raveendran, R. Mukundan, A comparative analysis of algorithms for fast computation of Zernike moments, Pattern Recognition 36 (3), pp. 731–742, 2003.

[147] Chee-Way Chonga, P. Raveendranb and R. Mukundan, Translation and scale invariants of Legendre moments, Pattern Recognition 37, pp. 119 – 129, 2004

[148] J. Rey-Debove et A. Rey. Le nouveau petit Robert. Dictionnaires Le Robert, 2004.

[149] A.R. Rao. A Taxonomy for Texture Description and Identification. Springer-Verlag, 1990.

[150] J.P. Cocquerez et S. Philipp. Analyse d'images : filtrage et segmentation. Editions Masson, 1995.

[151] S. Geman, D. Geman, et C. Graffigne. Locating texture and object boundaries. Pattern Recognition, 87 : pp. 165–177, 1987.

[152] S. Geman et C. Graffigne. Markov random fields and image models and their application to computer vision. Proc. of the International Congress of Mathematics, pages 1496–1517, 1987.

[153] A. Jain et F. Farrokhnia. Unsupervised texture segmentation using Gabor filters. Pattern Recognition, 24(12) : pp. 1167–1186, 1991.

[154] A. Drimbarean et P.F. Whelan. Experiments in colour texture analysis. Pattern Recognition, 22(10) : pp. 1161–1167, 2001.

[155] O. Le Cadet. Méthodes d'ondelettes pour la segmentation d'images. Application à l'imagerie médicale et au tatouage d'images. Thèse de doctorat, Institut National Polytechnique de Grenoble, Septembre 2004.

[156] L. Shi et B. Funt. Quaternion colour texture. AIC2005 Proc. 10th Congress of the International Color Association, pages 1-4, 2005.

[157] N. Le Bihan et S.J. Sangwine. Quaternion principal component analysis of color images. In IEEE. International Conference on Image Processing (ICIP), pages 809–812, 2003.

[158] R. Haralick, K. Shanmugan, et I. Dinstein. Textural features for image classification. IEEE Transactions on SMC, 3(6) : pp. 610–621, 1973.

[159] M. Sharma et S. Singh. Evaluation of texture methods for image analysis. In R. Linggard, editor, Proceedings of the 7th Australian and New Zealand Intelligent Information Systems Conference, pp. 117–121, 2001.

[160] M. Skrzypniak, L. Macaire, et J-G. Postaire. Indexation d'images de personnes par analyse de matrices de cooccurrences couleur. CORESA2000, Journées d'études et d'échanges Compression et représentation des signaux audiovisuels, 2000.

[161] D. Muselet. Reconnaissance automatique d'objets sous éclairage non contrôlé par analyse d'images couleur. Thèse de doctorat, Université des sciences et technologies de Lille1, Juillet 2005.

[162] Aude Oliva and Antonio Torralba. Modeling the shape of the scene : A holistic representation of the spatial envelope. International Journal of Computer Vision, 42 : pp. 145–175, 2001.

[163] Aude Oliva , Antonio Torralba, Building the gist of a scene: the role of global image features in recognition, Progress in Brain Research, pages 1-19, 2006.

[164] Hans G. Feichtinger, Thomas Strohmer: "Gabor Analysis and Algorithms", Birkhäuser, 1998.

[165] Olivier Augereau, Nicholas Journet, Jean-Philippe Domenger, "Reconnaissance et Extraction de Pièces d'identité : Une application industrielle à la détection de cartes d'identité et de passeports," 1re soumission à CIFED 2012, le 8 décembre 2011.

[166] L. VINCENT, Texture Segmentation Using Gabor Filters, Center For Intelligent Machines, McGill University, December 2000.

[167] Haykin S. Neural networks: a comprehensive foundation, 2nd Edition. Prentice Hall, New Jersey. 1999.

[168] E. Fix and J. Hodges FIX. Discriminatory analysis: Nonparametric discrimination: Consistency properties. Tech. Rep. 4, USAF School of Aviation Medicine, Randolph Field, Texas, 1951.

[169] G. Shakhnarovich, , T. Darrell, , P. Indyk. Nearest-Neighbor Methods in Learning and Vision: Theory and Practice. The MIT Press, March 2006.

[170] Richard O. Duda, Peter E. Hart, David G. Stork. Pattern Classification, Wiley Interscience, 2000.

[171] Warren S. McCulloch et Walter H. Pitts, A logical Calculus of the ideas immanent in nervous activity, Bulletin of Mathematical Biophysics, vol 5, pp. 115-133, 1943.

[172] D.O. Hebb, The organization of behavior, New York, Wiley, 1949.

[173] F. Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain, 1958.

[174] Riviere D., Mangin J., Papadopoulos D., Martinez J., Frouin V. et Regis J., Automatic Recognition of Cortical Sulci of the Human Brain Using a Congregation of Neural Networks, Medical Image Analysis,6, 2, pp: 77-92, Jun 2002.

[175] Marvin Minsky and Seymour Papert, Perceptrons: An Introduction to Computational Geometry, (2nd edition with corrections, first edition 1969) The MIT Press, Cambridge MA, ISBN 0-262-63022-2, 1972.

[176] John Joseph Hopfield, Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences, Vol.79, pp. 2554-2558, 1982

[177] Paul J. Werbos. Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences. PhD thesis, Harvard University, 1974.

[178] Paul J. Werbos. Back propagation through time: what it does and how to do it. Proceedings of the IEEE, Volume 78, Issue 10, pp. 1550 - 1560, Oct 1990.

[179] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. In D.E. Rumelhart and J. L. McClelland, editors, Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol.1: Foundations. Bradford Books/MIT Press, Cambridge, MA, 1986.

[180] Y. LeCun: Une procédure d'apprentissage pour réseau a seuil asymmetrique (a Learning Scheme for Asymmetric Threshold Networks), Proceedings of Cognitiva 85, 599–604, Paris, France, 1985.

[181] H. Zouari, "Contribution à l'évaluation des méthodes de combinaison parallèle de classifieurs par simulation", Thèse de Doctorat, Université de Rouen, 2004.

[182] Ripley B. D., Pattern Recognition and Neural Networks, Cambridge, Cambridge University Press, 1996.

[183] O. Lezoray. Segmentation d'images par morphologie mathématique et classification de données par réseaux de neurones :Application à la classification de cellules en cytologie des séreuses. Thèse de doctorat, Université de Caen 2000.

[184] Simard P., Steinkraus D., Platt J. C., Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis, ICDAR, pp: 958-962, 2005.

[185] R. Lepage, & B. Solaiman. Les réseaux de neurones artificiels et leurs applications en imagerie et en vision par ordinateur, Ecole de technologie supérieure, 2003.

[186] Vladimir Vapnik et A. Lerner, Pattern Recognition using Generalized Portrait Method, Automation and Remote Control, 1963.

[187] M. Aizerman, E. Braverman, and L. Rozonoer, Theoretical foundations of the potential function method in pattern recognition learning, Automation and Remote Control 25: pp. 821-837, 1964.

[188] Bernhard E. Boser, Isabelle M. Guyon, Vladimir N. Vapnik, A Training Algorithm for Optimal Margin Classifiers, In Fifth Annual Workshop on Computational Learning Theory, pages 144-152, Pittsburgh, ACM. 1992.

[189] V. Vapnik, The nature of statistical learning theory, N-Y, Springer-Verlag, 1995.

[190] B. Schölkopf , C.J.C. Burges et A.J. Smola. Advances in Kernel Methods: Support Vector Learning. MIT Press, 1999

[191] James Mercer, « Functions of positive and negative type and their connection with the theory of integral equations », Philos. Trans. Roy. Soc. London, 1re série, vol. 209, pp. 415-446, 1909.

[192] Cortes, C. et V. Vapnik, « Support-vector networks », Machine Learning, 20(3), pp. 273–297, 1995.

[193] Platt, J., Fast training of support vector machines using sequential minimal optimization, chapitre 12, pp. 185–208, C. J. C. Burges, and A. Smola editors, Advances in Kernel Methods : Support Vector Learning, MIT Press, 1999.

[194] R. Rifkin, A. Klautau. In defence of one-versus-all classificaiton. Journal of Machine Learning Research, Vol. 5, pp. 101–141, 2004.

[195] K.-B. Duan, S.S Keerthi, Which is the best multiclass SVM method? An empirical study. Technical Report CD-03-12, Control Division, Department of Mechanical Engineering, National University of Singapore, 2003.

[196] Thomas Bayes et Richard Price, « An Essay towards Solving a Problem in the Doctrine of Chances. By the Late Rev. Mr. Bayes, F. R. S. Communicated by Mr. Price, in a Letter to John Canton, A. M. F. R. S. », Phil. Trans., vol. 53, pp. 370-418, 1er janvier 1763.

[197] Judea Pearl, « Bayesian Networks: A Model of Self-Activated Memory for Evidential Reasoning », Proceedings of the 7th Conference of the Cognitive Science Society, UC Irvine, pp. 329–334, 1985.

[198] Judea Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann Publishers, Inc, San Francisco, USA, 1988 (ISBN 9781558604797).

[199] Perle De Judea, Stuart Russell. Réseaux Bayésiens. UCLA Laboratoire Cognitif De Systèmes, Rapport Technique (R-277), Novembre 2000. In M.A. Arbib (Ed.),

Handbook of Brain Theory and Neural Networks, Cambridge, MA: MIT Press, pp. 157-160, 2003.

[200] Pearl Judea. A constraint-propagation approach to probabilistic reasoning. In : L. N. Kanal and J. F. Lemmer (eds), Uncertainty in Artificial Intelligence, Amsterdam, NorthHolland, pages 3718–1986, 1986.

[201] Pearl Judea. Fusion, propagation and structuring in belief networks. UCLA Computer Science Department Technical Report 850022 (R-42) ; Artificial Intelligence,, 29 :241– 288, 1986.

[202] Pearl Judea and A. Paz. Graphoids : a graph-based logic for reasoning about relevance relations. UCLA Computer Science Department Technical Report 850038 ; In B. Du Boulay, et.al. (Eds.), Advances in Artificial Intelligence-II, North-Holland Publishing Co., 1987.

[203] Pearl Judea and T. Verma. Influence diagrams and d-separation. UCLA Cognitive Systems Laboratory, Technical Report 880052, 1988.

[204] D. Geiger, D. Heckerman, H. King, and C. Meek. Stratified exponential families : Graphical models and model selection. The Annals of Statistics, 29 : pp. 505–526, 2001.

[205] T. Bayes. An essay toward solving a problem in the doctrine of chance. Philosophical Transactions of the Royal Society, vol. 53, pages 370–418, 1763.

[206] Ann.Becker, Patrick Naim : les réseaux bayésiens : modèles graphiques de connaissance. Eyrolles.1999.

[207] J. Pearl, "Bayesian Networks" UCLA Cognitive Systems Laboratory, Technical Report (R-216), Revision I. In M. Arbib (Ed.), Handbook of Brain Theory and Neural Networks, MIT Press, 149-153, 1995.

[208] L.Smail : Algorithmes pour les réseaux bayésiens et leurs extensions. Thèse doctorat de l'Université de Polytech Nantes. Année 2004.

[209] Eduardo Sanchez Soto : Réseaux bayésiens dynamiques pour vérification du locuteur. Thèse doctorat 2005.

[210] Pascal Cheung- Mon- Chan : Réseaux bayésiens et filtres particuliers pour l'égalisation adaptative et le décodage conjoints, thèse doctorat de l'école normale supérieure de Cachan. Spécialité mathématique. Année 2006.

[211] F.Dress: Probabilités et statistiques de A à Z. 500 définitions, formules et tests d'hypothèse. Edition Dunod. paris.2004.

[212] Christian Robert. The bayesian Choice : a decision-theoretic motivation. Springer, New York, 1994.

[213] M.A.mahjoub, K.Jayech : New approach using Bayesian Network to improve content based image classification systems. IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 6, pp. 53-62, November 2010.

[214] George H. John and Pat Langley. Estimating continuous distributions in bayesian classifiers, the Eleventh Conference on Uncertainty in Artificial Intelligence, 1995.

[215] C.Aaron: Algorithme EM et classification non supervisée. Thèse de doctorat de l'Université Paris I. année 2001

[216] R. Robinson. Counting unlabeled acyclic digraphs. In C. Little, editor, Combinatorial Mathematics V, volume 622 of Lecture Notes in Mathematics, pages 28–43, Berlin, 1977. Springer.

[217] O.Francois, P.Leray : étude comparative d'algorithmes d'apprentissage de structure dans les réseaux bayésiens. Proceedings of the IEEE, vol. 60, no. 4, pp. 586–704, 2004.

[218] Z. Yun et K. Keong: Improved MDL scores for learning of Bayesian networks. Dans Proceedings of the International Conference on Artificial Intelligence in Science and Technology. AISAT 2004, pp. 98–103, 2004.

[219] C.Nicolas: Apprentissage de structure des réseaux bayésiens à partir de données incomplètes. Mémoire d'ingénieur en informatique spécialité réseaux et systèmes, 2008.

[220] Philippe LERAY, Réseaux bayésiens : apprentissage et modélisation de systèmes complexes, Habilitation à Diriger Les Recherches, Spécialité Informatique, Automatique et Traitement du Signal, Université de Rouen, novembre 2006.

[221] Patrick Naïm, Pierre Henri Wuillemin, Philippe Leray, Olivier pourret, Anna becker, Réseaux bayésiens, Eyrolles, 3ème édition, Paris, 2008.

[222] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. Machine Learning, 29(2-3) :pp. 131–163, 1997.

[223] Jie Cheng and Russell Greiner. Comparing bayesian network classifiers. In Proceedings of the Fifteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–99), pages 101–108, San Francisco, CA, Morgan Kaufmann Publishers, 1999.

[224] Jie Cheng and Russell Greiner. Learning bayesian belief network classifiers : Algorithms and system. In Proceedings of the Canadian Conference on AI 2001, volume 2056, pages 141–151, 2001.

[225] Tom .Mitchell: Generative and discriminative classifier: Naïve bayes and logistic regression. Machine learning. Draft 2010.

[226] Pat Langley, Wayne Iba, and Kevin Thompson. An analysis of Bayesian classifiers. In Proceedings of the Tenth National Conference on Artificial Intelligence, pages 223–228, San Jose, CA, AAAI Press, 1992.

[227] Y.Xin, Z.Zhaobao, Z.Haitao, Y. Zhiwei: Texture classification of aerial image based on bayesian network augmented naive bayes. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Vol. XXXVII. Part B7. Beijing 2008.

[228] E. Keogh and M. Pazzani. Learning augmented bayesian classifiers : A comparison of distribution-based and classification-based approaches. In Proceedings of the Seventh International Workshop on Artificial Intelligence and Statistics, pages 225–230, 1999.

[229] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. Machine Learning, 29(2-3) :pp. 131–163, 1997.

[230] J. Sacha, L. Goodenday, and K. Cios. Bayesian learning for cardiac spect image interpretation. Artificial Intelligence in Medecine, 26 :109–143, 2002.

[231] ETH-80 database image. Available Online: http://www.d2.mpi-inf.mpg.de/Datasets/ETH80

[232] COIL-100 database image. Available Online: http://www.cs.columbia.edu/CAVE/software/softlib/coil-100.php.

[233] M. Everingham, A. Zisserman, C. K. I. Williams, et L. Van Gool. The PASCAL Visual Object Classes Challenge 2006 (VOC2006) Results. http ://www.pascal-network.org/challenges/VOC/voc2006/results.pdf.

[234] Li Fei-Fei, Rob Fergus, et Pietro Perona. Learning generative visual models from few training examples : An incremental bayesian approach tested on 101 object categories. Computer Vision and Image Understanding, 106(1) :59 –70, 2007. Special issue on Generative Model Based Vision.

[235] G. Griffin, A. Holub, et P. Perona. Caltech-256 object category dataset. Rapport Technique 7694, California Institute of Technology, 2007.

[236] Y. Ait Ouguengay, M. Taalabi, "Elaboration d'un réseau de neurones artificiels pour la reconnaissance optique de la graphie amazighe: Phase d'apprentissage", Systèmes intelligents-Théories et applications, Paris: Europia, cop. 2009 (impr. au Maroc), ISBN-102909-285553, 2009.

[237] Mustapha OUJAOURA, R. EL AYACHI, O. BENCHAREF, Y. CHIHAB and B. JARMOUNI, application of data mining tools for recognition of tifinagh characters, (IJACSA) International Journal of Advanced Computer Science and Applications, special issue on selected papers from third international symposium on automatic amazigh processing SITACAM13, Vol. 3, No. 2, pp.1–4, 2013. Published by The Science and Information Organization, New York, USA.

[238] M. OUJAOURA, B. MINAOUI, M. FAKIR, B. BOUIKHALENE, R. EL AYACHI and O. BENCHAREF, Invariant Descriptors and Classifiers Combination for Recognition of Isolated Printed Tifinagh Characters, (IJACSA) International Journal of Advanced Computer Science and Applications, special issue on selected papers from third international symposium on automatic amazigh processing SITACAM13, Vol. 3, No. 2, pp.22–28, 2013. Published by The Science and Information Organization, New York, USA.

[239] M. OUJAOURA, R. EL AYACHI, B. MINAOUI, M. FAKIR and B. BOUIKHALENE, Zernike Moments and Neural Networks for Recognition of Isolated Arabic Characters, International Journal of Computer Engineering Science (IJCES), Volume 2 Issue 3, March 2012.

[240] Mustapha OUJAOURA, Brahim MINAOUI and Mohammed FAKIR. Article: Walsh, Texture and GIST Descriptors with Bayesian Networks for Recognition of Tifinagh characters. (IJCA) International Journal of Computer Applications, Volume 81– No.12, pp. 39-46, November 2013. Published by Foundation of Computer Science, New York, USA.

[241] M. OUJAOURA, B. MINAOUI, M. FAKIR, R. EL AYACHI and O. BENCHAREF, Recognition of Isolated Printed Tifinagh Characters, (IJCA) International Journal of Computer Applications, Volume 85 – No. 1 , pp.1 – 13 , January 2014. Published by Foundation of Computer Science, New York, USA.