Recherche d'images par le contenu pour la détection des copies

Robust Content-Based Image Searches for Copyright Protection

Sid-Ahmed Berrani[1] Laurent Amsaleg[2] Patrick Gros[2]

[1] Thomson Multimédia R&D France
1, av. Belle Fontaine – CS 17616. 35576 Cesson-Sévigné – France
Sid-Ahmed.Berrani@thomson.net

[2] IRISA – CNRS
Campus de Beaulieu. 35042 Rennes – France
{Laurent.Amsaleg, Patrick.Gros}@irisa.fr

Résumé

Cet article propose une nouvelle méthode de détection des copies d'une image basée sur une technique de recherche de ces copies par leur contenu. Son but est de rechercher, par exemple sur le Web ou dans un stock d'images de propriété douteuse, les images semblables à celles d'une collection d'images d'un propriétaire. Le système permet non seulement de détecter que les images douteuses ont été copiées, mais aussi de retrouver pour chacune d'entre elles de quelle image originale elle est la copie. L'emploi de descripteurs locaux permet de s'affranchir de nombreuses transformations que le copieur peut faire subir aux images afin d'en rendre l'identification plus difficile. Par contre, le coût élevé induit par ce schéma de reconnaissance nécessite l'emploi d'une technique de recherche très rapide car chaque recherche d'image se traduit par la recherche dans la base de nombreux descripteurs. Nous proposons donc d'utiliser une nouvelle technique d'indexation et de recherche qui, au prix d'une légère dégradation de la qualité des résultats, réduit fortement le temps de recherche. La dégradation des résultats induite par cette technique reste faible, car le résultat final est obtenu par accumulation de résultats partiels. Ce point fait l'objet dans l'article d'une étude théorique et d'une vérification expérimentale.

Mots Clef

Détection de copies, reconnaissance d'images, recherche approximative de plus proches voisins, invariants locaux différentiels.

Abstract

This paper proposes a novel content-based image retrieval scheme for copyright protection. Its goal is to detect matches between a set of doubtful images and the ones stored in the database of photographies legal holders. If an image was stolen and used to create a pirated copy, it tries to identify from which original image the copy was created. Thanks to local differential descriptors, the matching process takes into account a large set of variations that might have been applied to pirated images. The high cost and the complexity of this image recognition scheme requires a very efficient retrieval process since many individual queries must be performed before being able to construct the final result. This paper therefore proposes to use a novel search method that trades the precision of each individual search for reduced query execution time. This imprecision has only little impact on the overall recognition performance since the final result is a consolidation of many partial results. However, it dramatically accelerates queries. This result has then been corroborated by a theoretically study. Experiments show the efficiency and the robustness of the proposed scheme.

Keywords

Copyright protection, image recognition, approximate nearest neighbor searches, local differential invariants.

1 Introduction

La protection du droit d'auteur est un problème clé dont la solution est un préliminaire à la diffusion de collections de photos sur le Web par leurs propriétaires. Comme il n'est pas possible d'empêcher la copie d'images sur des sites Web par des personnes techniquement compétentes, les propriétaires d'images comme les agences de photos doivent disposer de moyens pour vérifier si les images présentes sur des sites tiers ne viennent pas de leur propre collection d'images. Ceci est d'autant plus important si le tiers tire profit des images qu'il prétend posséder, au détriment

de leur propriétaire légitime. Le tatouage d'images est la première solution à laquelle on peut penser, mais en faire un argument légal demande une organisation complexe sur le plan technique (de la technique elle-même) et organisationnel (des protocoles, des tiers de confiance...). Ceci est d'autant plus vrai que les images copiées sont souvent lessivées pour enlever les marques cachées ou rendre leur détection impossible.

Dans cet article, nous proposons une nouvelle méthode de détection de copies basée sur un schéma de reconnaissance d'images par le contenu. L'idée est de fournir aux propriétaires d'images des outils pour tester si une image provient de leur stock, et ce sur des bases visuelles seulement. Le but d'un tel système est alors d'être capable d'établir des liens de ressemblances entre des images issues du Web et celles d'une collection possédée par un propriétaire et, au cas où une telle ressemblance existe, de trouver quelle est l'image de la collection qui a été copiée.

Dans un tel contexte, les systèmes de reconnaissance d'images par le contenu sont soumis principalement à trois contraintes.

Robustesse du schéma de description. Les systèmes d'indexation et de recherche par le contenu (systèmes CBIR) doivent être capables, dans un contexte de détection des copies, de reconnaître une image copiée à partir de l'original, même si la copie a été modifiée : il est probable que les pirates modifient les images qu'ils récupèrent (par exemple pour enlever un possible tatouage). Le système CBIR utilisé doit en conséquence être robuste à certaines modifications : changement de taille, modification légère des couleurs ou du contraste, rotation, compression JPEG, symétrie miroir, extraction d'une sous-image, etc. Il ne s'agit pas, dans un tel contexte, de reconnaître toutes les images de fleurs rouges à partir d'une image requête représentant elle aussi une fleur rouge, mais plutôt de déterminer si une image suspecte est une copie d'une image que l'on détient, et dans l'affirmative, de retrouver de quelle image elle en est une copie. Cela impose d'utiliser une notion de similarité bien plus contraignante et fine que celles utilisées par la plupart des systèmes CBIR. On parlera dans la suite d'images semblables plutôt que d'images similaires pour souligner cette différence.

Efficacité de la recherche. Les systèmes CBIR ne peuvent pas fournir de preuves légales de copie, ils ne peuvent qu'émettre des suspicions. La preuve finale du fait qu'il y a eu copie ne peut être fournie que par un expert humain à qui sont transmis les résultats obtenus par le système. Il importe donc que ce dernier ait d'excellents rappel et précision : rappel, car le but du système est bien de détecter les copies et il ne faut pas en rater, précision, car tout faux-positif se traduit par un travail de l'expert dépensé en vain. Le système doit donc fournir des résultats de très haute qualité.

Rapidité de la recherche. Dans le cadre de la détection des copies, un système CBIR doit pouvoir traiter un grand nombre d'images récupérées sur le Web et les confronter à de grandes bases d'images. La réponse à une requête doit donc être rapide. Or, les systèmes CBIR sont le plus souvent basés sur l'emploi de techniques d'indexation multidimensionnelles pour réaliser des recherches de plus proches voisins : de tels algorithmes sont déjà complexes, mais la fameuse « malédiction de la dimension » [1] renforce ce phénomène en rendant toute recherche de plus proches voisins dans un espace de grande dimension intrinsèquement complexe, indépendamment de la technique d'indexation et de recherche utilisée. L'aspect rapidité du système apparaît donc à la fois comme crucial et potentiellement bloquant.

Le système que nous proposons dans cet article prend en compte les trois critères décrits ci-dessus. Il utilise une technique robuste de description du contenu visuel des images. Cette technique est basée sur l'extraction de points d'intérêt et la caractérisation du signal autour de ces points par des invariants différentiels locaux [2]. Le système utilise alors une technique efficace de recherche approximative de plus proches voisins pour accélérer les recherches. L'approximation faite tend à échanger un peu de précision dans les résultats contre une accélération notoire du temps de recherche [3].

La première contribution du présent article est la combinaison de ces deux méthodes, celle de description locale et celle de recherche approximative, rendant possible le développement d'un système de détection des copies basé sur une reconnaissance des images copiées par le contenu. Un des points importants est de mesurer l'impact en termes de qualité des résultats de l'approximation réalisée lors de la recherche, et de vérifier que cette qualité reste acceptable pour l'application que nous visons. La deuxième contribution de l'article est donc une vérification de cela, tant théorique que pratique.

L'article est organisé de la manière suivante. Dans la section 2 est discuté le problème de la description des images dans le cadre de la détection de copies et la méthode que nous avons retenue est brièvement présentée. La méthode de recherche approximative est présentée à la section 3. La section 4 est consacrée à l'étude de l'influence de l'approximation faite pendant la recherche sur la qualité des résultats de cette même recherche. La section 5 présente les résultats expérimentaux, en termes de rapidité et de qualité et illustre en pratique l'étude de la section précédente. Les résultats de rapidité obtenus sont comparés à ceux que l'on peut obtenir avec une recherche séquentielle et exhaustive, qui est la méthode étalon du domaine : c'est une méthode simple, trop coûteuse face à de très grandes bases de données, mais que très peu de méthodes arrivent à battre [1].

2 Reconnaissance d'images par le contenu

La reconnaissance d'images par le contenu est basée sur la calcul de quantités décrivant les images, quantités appelées descripteurs, et sur l'hypothèse que la mesure de la ressemblance entre les images peut se ramener à celle de la distance entre les descripteurs calculés à partir de ces images. Une grande variété de descripteurs a été proposée dans la littérature [4]. Le schéma de description le plus fréquent est celui des descripteurs globaux où un unique descripteur décrit la globalité d'une image. Les histogrammes de couleur ou de niveaux de gris et les corrélogrammes sont des exemples typiques de tels descripteurs. Ces descripteurs sont robustes au sens où ils sont peu affectés par le bruit qui peut affecter le signal. En contrepartie, ils ne sont pas adaptés pour retrouver des parties communes entre deux images et ne sont pas robustes en cas de changement de résolution des images ou d'extraction de sous-images, propriété qui est indispensable dans le contexte de la détection des copies.

À l'opposé, les méthodes de reconnaissance basées sur des descripteurs locaux offrent naturellement ces propriétés d'invariance, et paraissent donc plus adaptées à notre contexte applicatif. Ces méthodes consistent à calculer de nombreux descripteurs pour une même image, chaque descripteur décrivant une petite partie seulement de l'image, que ce soit autour d'un point ou dans une petite région de l'image. La reconnaissance globale entre deux images est alors jugée en fonction du nombre de descripteurs locaux qui coïncident, à ε près, entre les deux images. Un tel schéma prend en compte les détails des images et permet la prise en compte d'une notion fine de similarité entre les images. On parlera d'images semblables dans ce cas. Ce schéma offre également une bonne robustesse naturellement car il se base sur l'accumulation d'évidences locales.

Dans le système que nous présentons ici, nous avons retenu ce deuxième mode de description. Les descripteurs utilisés appartiennent à la famille des invariants différentiels locaux introduite par Florack [5] puis utilisée et évaluée dans le contexte de la reconnaissance d'images en niveaux de gris par Schmid *et al.* [6, 7]. Leur extension à la couleur a été présentée dans [8, 2]. Ces descripteurs offrent une très grande robustesse aux transformations qu'une image peut subir, et ils peuvent détecter que deux images ont une partie en commun seulement, et cela à une translation, rotation, variation géométrique ou photométrique près.

Le calcul de ces descripteurs locaux pour décrire le contenu d'une image est réalisé en trois étapes. (i) On commence par détecter dans l'image des points particuliers, dits points d'intérêts, qui portent une partie importante de l'information contenue dans le signal. Ces points correspondent à des singularités du signal et sont détectés en utilisant l'algorithme de Harris [9] amélioré par Schmid [6]. Le nombre de points détectés dans une image peut varier, car il dépend de la forme du signal de l'image et de la technique de seuillage utilisée ; (ii) le signal autour de chacun de ces points d'intérêt est caractérisé par sa convolution avec une gaussienne et ses 9 premières dérivées jusqu'à l'ordre 3 ; (iii) enfin, ces 10 quantités obtenues sont combinées afin d'obtenir les propriétés d'invariance souhaitées en fonction de l'application visée. Pour le présent article, nous avons utilisé les descripteurs présentés dans [2] : ce sont des descripteurs de dimension 24 qui sont invariants aux translations, rotations et changement d'illumination décrits en RGB par une matrice diagonale et un vecteur de translation.

La recherche dans la base des images qui sont semblables à une image requête est basée sur la recherche des k plus proches voisins de chacun des descripteurs locaux de l'image requête. Chacun de ces plus proches voisins obtenus donne un vote à l'image de la base dont il est issu. Les images de la base peuvent alors être classées selon le nombre de votes qu'elles ont obtenu, nombre noté NDS (nombre de descripteurs semblables) dans la suite. NDS donne le nombre de descripteurs en commun (ou proches) entre cette image et l'image requête.

De manière générale, de très nombreuses images reçoivent au moins un vote, voire quelques votes. Si la base ne contient aucune image semblable à l'image requête, toutes les images qui reçoivent des votes ont des NDSs voisins et faibles. À l'opposé, si la base contient des images semblables, la liste des NDSs obtenus se sépare clairement en deux groupes : les images semblables obtiennent des scores élevés qui décroissent en fonction de la ressemblance de ces images avec l'image requête, puis il y a un saut et on trouve ensuite les autres images qui ont des scores faibles. Voici un exemple typique d'une telle liste de NDSs ordonnée de manière décroissante : $444, 258, 142, 42, 17, 14, 10, 9..., 9, 8..., 8, 7...$

Il y aurait un grand intérêt à pouvoir déterminer *automatiquement* quelles sont les images semblables à l'image requête. Mais cela n'est pas immédiat. Prendre un nombre fixe d'images (la première, les cinq ou les seize premières en fonction de la taille de l'écran d'affichage) ne permet pas de prendre en compte la redondance possible dans la base et peut dégrader la précision. Par ailleurs, cela implique que le système retournera toujours une réponse et ne peut pas déclarer qu'aucune image n'est semblable à une image donnée. Si toutes les images sont envoyées à un expert et que l'on teste des milliers d'images par jour, cela devient inacceptable.

L'utilisation d'un seuil fixe est aussi problématique car l'ordre de grandeur des NDSs dépend du nombre des descripteurs requêtes d'une part, et aussi de la zone commune entre l'image requête et l'image recherchée d'autre part. Il n'est donc pas possible de fixer *a priori* un seuil qui puisse marcher pour une base donnée indépendamment de l'image requête. La solution que nous proposons est d'assimiler ce problème à celui de la détection d'une transition dans un signal et d'utiliser une technique issue du traitement du signal pour détecter cette transition. Pour cela, on trie la

liste des NDSs par ordre croissant et on lui applique le test à somme cumulée de Page-Hinkley [10, 11, 12]. Ce test est connu pour sa robustesse, car il permet de prendre en compte l'intégralité du passé de la liste à un instant donné. Par ailleurs, il est efficace et peu coûteux, il permet une détection précise de la position du saut, et il peut traiter à la fois les changements abrupts et les changements progressifs sans modification du seuil interne de détection de la méthode. Le système retourne finalement les images qui ont été séparées des autres par ce test, ou un résultat vide si le test n'a détecté aucune transition dans la liste des NDSs.

Ce schéma de description, s'il apporte finesse et précision dans la recherche, modifie toutefois en profondeur la méthode de recherche des images semblables. Tout d'abord, il augmente la taille de la base de données : au lieu de stocker un unique descripteur par image, il nécessite d'en stocker un nombre qui peut varier entre 10 et 600 environ pour des images 512×512. La base grossit typiquement d'un facteur 200. Ensuite, ce schéma de description modifie l'algorithme de recherche lui-même. En effet, il ne s'agit plus de rechercher les descripteurs les plus proches d'un unique descripteur requête, mais de mener de nombreuses recherches à partir de tous les descripteurs de l'image requête, induisant autant de recherche de plus proches voisins dans la base qu'il y a de descripteurs requêtes, puis de combiner ces résultats partiels pour obtenir un résultat final.

Le coût de la recherche de plus proches voisins étant intrinsèquement élevé, cela pose un problème de complexité de la méthode et nécessite l'emploi d'une méthode extrêmement rapide. À l'inverse, la méthode de description reposant sur une redondance de résultats partiels, il est possible d'autoriser une légère diminution de la qualité des résultats partiels sans dégrader le résultat final, si cette dégradation permet un fort gain en temps de calcul. Cela fait l'objet de la section suivante.

3 Recherche par similarité

De nombreuses méthodes de recherche de plus proches voisins (ppv) ont été proposées [13]. Il s'agit des index multidimensionnels qui permettent de structurer les données de la base souvent de manière arborescente et d'y associer des algorithmes de recherche. Plusieurs travaux ont montré que les performances de ces techniques se dégradent exponentiellement lorsque la dimension des descripteurs augmente ($>$16 selon [14]). Les descripteurs d'images étant généralement de dimension très élevée, ces techniques ne sont pas capables d'accélérer les recherches dans les bases d'images. Les techniques de recherche approximative sont par contre plus efficaces et gardent de bonnes performances même lorsque la dimension augmente. Elles réduisent considérablement le coût de la recherche mais ne permettent pas de retrouver les ppv exacts d'un vecteur requête. Elles se contentent de retourner un ensemble de vecteurs parmi les voisins du vecteur requête mais pas forcément les plus proches. Actuellement, la recherche approximative semble être la seule solution qui permet de pallier les problèmes liés aux espaces de grande dimension et d'effectuer des recherches efficaces dans ces espaces [15, 16, 3].

Par ailleurs, dans le contexte de la recherche d'images par descripteurs locaux, le résultat final est construit à partir de plusieurs résultats intermédiaires. Il se déduit sur la base de l'accumulation de plusieurs évidences locales. Il est, par conséquent, moins sensible à l'imprécision introduite par la recherche de ppv. Pour toutes ces raisons, nous avons donc choisi la recherche approximative pour accélérer les recherches.

Un panorama des techniques de recherche approximative est donné dans [13]. Nous avons choisi d'utiliser la méthode proposée par Berrani *et al.* dans [3, 17]. L'originalité de cette méthode est sa capacité à contrôler la précision de la recherche de manière probabiliste, en ligne et à l'aide d'un seul paramètre facile à fixer, α, appelé niveau d'imprécision. α est la probabilité maximale de ne pas retrouver un des k ppv exacts recherchés dans le résultat approximatif. En d'autres termes, cette méthode renvoie un résultat approximatif tout en garantissant que la probabilité de ne pas retrouver un des k ppv exacts du vecteur requête soit inférieure ou égale à α.

Hors ligne, cette méthode effectue un *clustering* des vecteurs de la base et isole les vecteurs éloignés *(outliers)*. Pour chaque *cluster*, elle calcule le rayon exact (la distance entre le centre de gravité et son plus lointain voisin parmi les vecteurs du *cluster*) et un ensemble de rayons approximatifs. Chaque rayon approximatif correspond à un niveau d'imprécision α et est calculé tel que la probabilité d'ignorer un des ppv exacts soit inférieure à α si ce ppv appartient au *cluster*.

Ces rayons définissent des hypersphères englobantes qui sont utilisées lors de la recherche (la phase en ligne) pour filtrer les *clusters* non pertinents en raisonnant uniquement sur les distances minimales et maximales entre le vecteur requête et les hypersphères englobantes. Si un *cluster* est filtré, les vecteurs qu'il contient ne sont pas chargés en mémoire et les distances entre ses vecteurs et le vecteur requête ne sont pas calculées. Ce qui permet ainsi de réduire le temps de recherche. Le fichier des vecteurs isolés est, quant à lui, parcouru séquentiellement.

Nous avons aussi utilisé une heuristique permettant d'accélérer la recherche en réduisant le coût des calculs de distances. Il s'agit d'interrompre le calcul de la distance dès que la distance partielle calculée sur un sous-ensemble des composantes des vecteurs est supérieure à la distance au k^e ppv courant. Dans ce cas, le vecteur en cours ne peut être plus proche du vecteur requête que le ppv courant, et par conséquent, il est inutile de poursuivre le calcul de la distance.

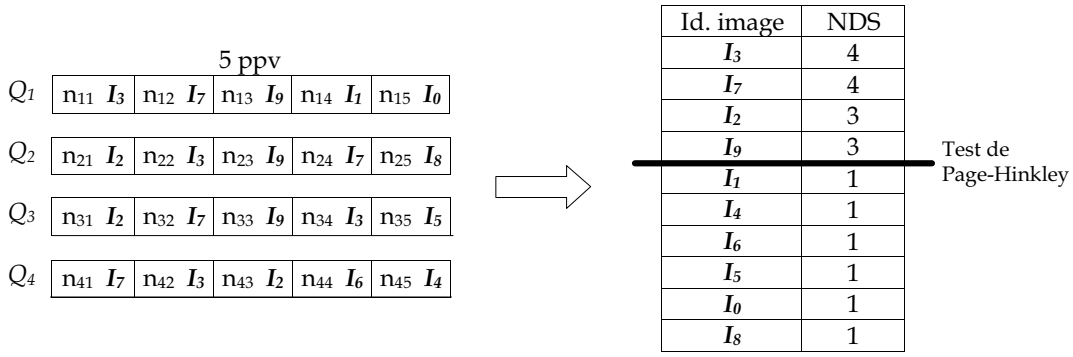


FIG. 1 – *Exemple illustratif de la procédure de vote.*

4 Précision de la recherche et descripteurs locaux

Comme nous l'avons expliqué dans la section précédente, il est clair que l'imprécision de la recherche de ppv n'a qu'un impact mineur sur le résultat final de la recherche d'images par descripteurs locaux. Néanmoins, il est important d'étudier de manière précise la relation entre l'imprécision de la recherche de ppv effectuée individuellement pour chaque vecteur requête et l'imprécision du résultat final. Avant de présenter cette étude, nous détaillerons d'abord la procédure de vote qui permet de déterminer et d'ordonner les images semblables à l'image requête.

Nous considérons une image requête I_q décrite à l'aide de n descripteurs (Q_1,...,Q_n). Pour chacun de ces descripteurs, k ppv approximatifs ont été retrouvés par l'algorithme de recherche approximative. Le niveau d'imprécision a été fixé à α. Chaque ppv d'un descripteur requête attribue un vote à l'image à laquelle il appartient. À la fin de la recherche, c'est-à-dire, une fois que tous les ppv de tous les vecteurs requêtes ont été retrouvés, à chaque image est associé le nombre de votes qu'elle a reçu. Ce nombre reflète le degré de similarité entre l'image retrouvée et l'image requête. Il est noté NDS (Nombre de Descripteurs Similaires). L'ensemble des NDSs est ensuite ordonné et seuls les NDSs significatifs sont gardés après application du test de Page-Hinkley. Les images dont les NDSs sont supérieurs au seuil déterminé par le test de Page-Hinkley constituent le résultat final.

La figure 1 illustre la procédure de vote pour une image requête décrite à l'aide de quatre descripteurs (Q_1,Q_2,Q_3, Q_4). Pour chaque descripteur requête, 5 ppv ont été retrouvé. Dans cette figure, à chacun de ppv est associé l'identifiant de l'image à laquelle il appartient. Comme nous l'avons expliqué précédemment, il suffit de compter le nombre de votes par image pour construire la liste des résultats. La partie droite de la figure présente cette liste pour cet exemple. Dans ce cas là, appliquer le test de Page-Hinkley à la liste des NDSs permet d'isoler les images similaires du reste : les images ayant un NDS $\geqslant 3$ sont considérées comme similaires.

Les recherches de ppv étant effectuées par un algorithme de recherche approximative, la probabilité de ne pas retrouver l'un des k ppv exacts est inférieure à α. Ne pas retrouver un des k ppv exacts d'un vecteur requête modifie les NDSs des images semblables dans le résultat final. En termes plus précis, ignorer un des k ppv exacts d'un vecteur requête permet de :

- décrémenter de 1 le NDS de l'image à laquelle appartient le ppv ignoré et
- incrémenter de 1 le NDS de l'image décrite par le ppv approximatif introduit en remplacement du ppv ignoré.

Cette modification des scores peut entraîner des inversions dans l'ordre des images semblables qui peut nuire plus ou moins gravement à la qualité du résultat final.

La probabilité d'inverser deux images semblables ordonnées successivement dans la liste du résultat final est liée à la différence entre les nombres de votes qui leurs ont été attribués : plus cette différence est grande et plus la probabilité de les inverser est petite. Pour clarifier ce point, nous considérons deux images similaires I_1 et I_2, ordonnées successivement dans la liste du résultat final et ayant reçu respectivement NDS_1 et NDS_2 avec une recherche de ppv exacte. Ces deux images ne peuvent être inversées si la recherche est effectuée par un algorithme de recherche approximative que si :

- au moins δ_1 descripteurs ayant voté pour l'image I_1 avec un algorithme de recherche exacte n'ont pas été retrouvés par la recherche approximative (évènement E_1), et,
- au moins δ_2 descripteurs de l'image I_2 ont été introduits par la recherche approximative alors qu'ils ne faisaient pas partie des ppv lors de la recherche exacte (évènement E_2),

tel que $\delta_1 + \delta_2 > \mathrm{NDS}\,1 - \mathrm{NDS}_2$.

La probabilité d'inverser les deux images I_1 et I_2 est donc $P(E_1).P(E_2)$.

Concernant l'évènement E_1, la probabilité de ne pas retrouver δ_1 parmi les k ppv des descripteurs requêtes est de α^{δ_1} (un ppv exact n'a pas été retrouvé pour un sous-ensemble δ_1 de l'ensemble des descripteurs requêtes). Les descripteurs requêtes sont supposés être indépendants étant donné qu'ils sont calculés indépendamment sur des points différents de l'image.
La probabilité de E_2 est, cependant, plus difficile à estimer. Les δ_2 descripteurs introduits par la recherche approximative peuvent être divisés en deux sous-ensembles. Un premier sous-ensemble de vecteurs est introduit en remplacement d'une partie des δ_1 ppv exacts de l'image I_1 qui n'ont pas été retrouvés. L'autre sous-ensemble contient les ppv approximatifs introduits en remplacement de ppv exacts appartenant à d'autres images. Posons γ le nombre de vecteurs de ce dernier sous-ensemble. $P(E_2)$ est donc égale à α^{γ}. Les ppv approximatifs appartenant au premier sous-ensemble n'ont pas été pris en compte dans le calcul de $P(E_2)$ étant donné qu'ils ont remplacé une partie des δ_1 ppv exacts de l'image I_1. Ils ont donc déjà été pris en compte dans le calcul de $P(E_1)$.

Ainsi, la probabilité d'inverser les images I_1 et I_2 dans le résultat final est $\alpha^{\delta_1+\gamma}$. Les valeurs de δ_1 et γ dépendent de la différence entre les NDSs des images I_1 et I_2 : plus cette différence est grande, plus grands sont δ_1 et γ et plus petite est la probabilité d'inverser les images I_1 et I_2 dans le résultat final. Malheureusement, aucune estimation théorique des valeurs de δ_1 et γ en fonction de α n'est possible. Néanmoins, cette étude a permis de montrer clairement que la probabilité d'inverser deux images dans le résultat final est reliée exponentiellement à α. Par conséquent, elle est beaucoup plus petite que la probabilité de ne pas retrouver un des ppv exacts d'un vecteur requête. L'utilisateur peut donc fixer le paramètre α à une valeur très élevée sans que cela ne détériore significativement le résultat final de la recherche.

5 Évaluation des performances

Pour conduire nos expériences, nous avons utilisé une base contenant 9544 images, parmi lesquelles 610 images de la base MOVI [1]. Les 9544 images ont été décrites à l'aide de 2 623 824 invariants différentiels locaux (descripteurs de dimension 24). En moyenne, chaque image a été décrite par 275 descripteurs.
Tous les algorithmes ont été implémentés en C++. Ils ont été exécutés sur un PC sous Linux disposant d'un processeur Intel Pentium 4 cadencé à 2,4 GHz et de 1 giga octets de mémoire centrale. Les temps d'exécution des algorithmes de recherche reportés dans la suite ont été obtenus grâce à la fonction `getrusage()`.

1. Disponible en suivant l'URL : http://www.irisa.fr/texmex/base_images/index.html

5.1 Expérience 1 : robustesse et efficacité

Afin d'étudier les performances du système de recherche d'images pour la détection des copies proposé dans cet article, nous avons mené un ensemble d'expériences afin :

- d'évaluer la capacité de reconnaissance des invariants différentiels locaux et leur robustesse face aux variations que peuvent subir les images piratées (reproduites illégalement) ;
- d'évaluer l'efficacité de la stratégie de recherche de ppv employée ;
- d'étudier l'impact de l'imprécision de la recherche de ppv sur la qualité des résultats finaux ;

Pour cela, nous avons utilisé les 32 séquences de la base MOVI. Chaque séquence représente la même scène prise dans des conditions différentes. Chaque séquence est utilisée pour étudier un changement particulier. Parmi ces séquences, on retrouve par exemple des images représentant la même scène prises dans des conditions d'illumination différentes (changement de l'intensité lumineuse, variation de la composition ou de la position de la source lumineuse, etc.). D'autres séquences ont été générées en modifiant la composition de la scène ou en déplaçant la caméra autour de la scène selon des mouvements plus ou moins complexes. L'ensemble de ces séquences sont décrites dans [2].
Notre protocole d'évaluation procède de la manière suivante. Il choisit une image à partir de chaque séquence et l'utilise pour interroger la base. Il garde ensuite pour chaque interrogation le temps de réponse et calcule le taux de précision et le taux de rappel du résultat retourné. Dans notre cas, la vérité terrain pour une image requête est parfaitement définie et est constituée de l'ensemble des images qui appartiennent à la même séquence que l'image requête. Une autre mesure de performance très utilisée dans le domaine de la recherche d'information permettant de combiner la précision (P) et le rappel (R) est aussi calculée [18]. Il s'agit de la F-Mesure (*cf.* équation 1).

$$\text{F-Mesure} = \frac{2\,\text{R}\,\text{P}}{\text{R}+\text{P}} \tag{1}$$

Afin d'illustrer ce protocole d'évaluation, nous présentons tout d'abord une expérience d'interrogation à l'aide d'une seule image requête. Cette image est tirée d'une séquence représentant la même scène prise sous différentes intensités lumineuses (*cf.* tableau 1). Cette image a été recherchée selon la procédure décrite dans la section 2. La recherche de ppv a été exécutée par un algorithme de recherche séquentielle de base, un algorithme de recherche séquentielle optimisée (proposé dans [19]) et par l'algorithme de recherche approximative avec différents degrés d'imprécision (α = 0,01, 0,10, 0,20, 0,40).
Les résultats obtenus sont résumés dans le tableau 2. Ils montrent la capacité des invariants locaux différentiels utilisés à absorber ce type de variation. En effet, toutes les images appartenant à la même séquence que l'image requête

TAB. 1 – *Exemple d'une image requête et deux de ses images semblables.*

Algo. de recherche		Temps de réponse (s)	NDS								# Faux nég.	# Faux pos.
			A_5	A_3	A_6	A_2	A_1	A_7	A_8	A_9		
Seq. de base		285,77	613	517	482	361	218	216	103	27	0	0
Seq. optimisée		127,98	613	517	482	361	218	216	103	27	0	0
α	0,01	41,52	617	515	481	361	220	211	102	26	0	6
	0,10	29,21	595	495	455	349	207	201	94	24	0	3
	0,20	25,39	542	433	401	312	182	180	83	22	0	5
	0,40	22,82	381	315	292	251	142	132	64	16	0	3

TAB. 2 – *Résultats de la recherche : temps de réponse et NDSs des images semblables retrouvées pour différentes stratégies de recherche.*

Algorithme de recherche	Temps de rep. moy. (s)	% moy. de reduc. du temps de rep. par rap. au seq. de base	Precision moy.	Rappel moy.	F-Mesure moy.
Seq. de base	191,31	-	0,65	0,93	0,77
Seq. optimisé	88,11	53,94	0,65	0,93	0,77
Rech. approx. ($\alpha = 0.01$)	27,50	85,63	0,64	0,92	0,75
Rech. approx. ($\alpha = 0.10$)	19,35	89,89	0,65	0,91	0,75
Rech. approx. ($\alpha = 0.20$)	17,05	91,09	0,65	0,90	0,75
Rech. approx. ($\alpha = 0.40$)	15,50	91,90	0,65	0,89	0,75

TAB. 3 – *32 images requêtes : précision moyenne, rappel moyen et accélération moyenne pour différentes stratégies de recherche.*

ont été retrouvées. Ils montrent aussi que la recherche approximative n'a qu'un impact mineur sur la qualité du résultat final alors qu'elle permet de réduire considérablement le temps de réponse (un facteur d'accélération de 6,88 (resp. 12,52) a été observé par rapport à la recherche séquentielle de base lorsque $\alpha = 0{,}01$ (resp. 0,40)). Quelques faux positifs ont été introduits. Ils ont été cependant ordonnés après les images pertinentes dans la liste des images semblables.

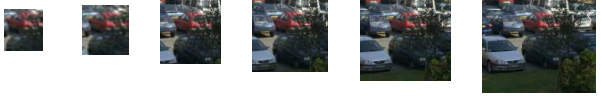
Nous avons ensuite effectué cette même expérience pour l'ensemble des 31 séquences restantes de la base MOVI afin de tester la robustesse du système par rapport à l'ensemble des variations. Les moyennes des résultats obtenus sont résumées dans le tableau 3. Les taux moyens de rappel et de précision sont tous les deux élevés, ce qui montre bien la grande capacité de ces descripteurs à reconnaitre même les images ayant subies des transformations géométriques et photométriques complexes. En moyenne, 92% des images semblables aux images requêtes sont toujours retrouvées. Elles représentent 65% de l'ensemble des images retournées. Par ailleurs, il est à noter que, pour 23 images requêtes sur l'ensemble des 32, toutes les images semblables ont été retrouvées (rappel $= 1$).

Ces résultats ont permis aussi de corroborer l'étude de l'impact de l'imprécision de la recherche de ppv sur le résultat final présenté dans la section 4. La recherche approximative a permis de réduire considérablement le temps de recherche en sauvegardant la qualité du résultat final par rapport à la recherche séquentielle. Des réductions du temps de recherche de l'ordre de 85 à 91% ont été réalisées par rapport à la recherche séquentielle de base tout en gardant à peu près le même taux de reconnaissance que la recherche séquentielle (une légère diminution de la F-Mesure, 0,75 contre 0,77 pour la recherche exacte).

5.2 Expérience 2 : robustesse au recadrage

Souvent les images piratées sont recadrées avant d'être publiées. Aussi, un système CBIR pour la détection des copies doit être capable de reconnaître une image à partir de petits fragments de celle-ci. Dans cette section, nous présentons une expérience qui permet d'étudier les performances des invariants locaux différentiels face à ce type « d'attaque ». Nous étudions aussi l'impact de l'imprécision de la recherche de ppv sur le résultat final.

Pour cela, nous avons sélectionné aléatoirement 20 images de la base (qui contient 9544 images). Six sous-images ont été extraites à partir de chacune des 20 images sélectionnées. Les proportions des surfaces des sous-images extraites par rapport à la surface des images originales sont 6,25, 9, 16, 25, 36 et 56,25%. Ces fragments ont été utilisés pour interroger la base. Le tableau 4 présente un exemple d'une image et les six fragments (ou sous-images) associés.

Image originale	Fragments requêtes associés
	 6,25% 9% 16% 25% 36% 56,25%

TAB. 4 – *Exemple d'une image et les six fragments requêtes associés*

% de l'image orig.	Taux de reconnaissance			
	Rech. exacte	Rech. approx (α)		
		0,01	0,20	0,40
6,25	0,94	0,94	0,94	0,94
9	0,97	0,97	0,97	0,97
16	0,97	0,97	0,97	0,97
25	1	1	1	0,97
36	1	1	1	1
56,25	1	1	1	1

(a) Fragments originaux

% de l'image orig.	Taux de reconnaissance			
	Rech. exacte	Rech. approx (α)		
		0,01	0,20	0,40
6,25	0,93	0,93	0,93	0,88
9	0,95	0,95	0,95	0,90
16	0,98	0,97	0,98	0,98
25	1	1	1	0,98
36	1	1	1	1
56,25	1	1	1	1

(b) Fragments + filtre gaussien

% de l'image orig.	Taux de reconnaissance			
	Rech. exacte	Rech. approx (α)		
		0,01	0,20	0,40
6,25	0,86	0,86	0,83	0,76
9	0,93	0,90	0,88	0,83
16	0,98	0,98	0,93	0,86
25	0,98	0,98	0,93	0,90
36	0,98	0,98	0,98	0,93
56,25	1	1	1	0,98

(c) Fragments + filtre médian

% de l'image orig.	Taux de reconnaissance			
	Rech. exacte	Rech. approx (α)		
		0,01	0,20	0,40
6,25	0,55	0,55	0,50	0,30
9	0,90	0,90	0,90	0,65
16	0,90	0,90	0,90	0,85
25	0,90	0,90	0,90	0,90
36	1	1	1	1
56,25	1	1	1	1

(d) Fragments + Rotation (2^o)

% de l'image orig.	Taux de reconnaissance			
	Rech. exacte	Rech. approx (α)		
		0,01	0,20	0,40
6,25	0,50	0,45	0,40	0,20
9	0,75	0,75	0,65	0,40
16	0,90	0,90	0,90	0,85
25	0,90	0,90	0,90	0,90
36	0,95	0,95	0,95	0,90
56,25	1	1	1	1

(e) Fragments + Rotation (10^o)

% de l'image orig.	Taux de reconnaissance			
	Rech. exacte	Rech. approx (α)		
		0,01	0,20	0,40
6,25	0,10	0,10	0,05	0
9	0,35	0,35	0,15	0,15
16	0,70	0,70	0,65	0,50
25	0,90	0,90	0,75	0,50
36	0,95	0,95	0,85	0,75
56,25	1	1	0,95	0,90

(f) Fragments + Rotation (45^o)

% de l'image orig.	Taux de reconnaissance			
	Rech. exacte	Rech. approx (α)		
		0,01	0,20	0,40
6,25	0,45	0,50	0,35	0,15
9	0,50	0,50	0,40	0,25
16	0,65	0,65	0,55	0,30
25	0,70	0,70	0,65	0,50
36	0,95	0,90	0,90	0,85
56,25	1	1	0,90	0,80

(g) Fragments + *Sharpening*

% de l'image orig.	Taux de reconnaissance			
	Rech. exacte	Rech. approx (α)		
		0,01	0,20	0,40
6,25	0,75	0,75	0,60	0,10
9	0,85	0,85	0,75	0,70
16	0,90	0,90	0,85	0,80
25	0,95	0,95	0,95	0,95
36	1	1	1	1
56,25	1	1	1	1

(h) Fragments + *Shearing*

% de l'image orig.	Taux de reconnaissance			
	Rech. exacte	Rech. approx (α)		
		0,01	0,20	0,40
6,25	0,61	0,60	0,52	0,36
9	0,78	0,78	0,72	0,63
16	0,88	0,88	0,86	0,79
25	0,92	0,92	0,89	0,83
36	0,98	0,98	0,96	0,94
56,25	0,99	0,99	0,98	0,97

(i) Résultats moyens sur toutes les variations

TAB. 5 – *Taux de reconnaissance moyen pour différentes tailles de fragments requêtes et différentes stratégies de recherche.*

Nous avons appliqué ensuite 5 variations différentes sur chaque fragment requête. Il s'agit :

- d'un renforcement de contraste *(Sharpening)* ;
- d'un cisaillement *(Shearing)* ;
- de rotations de 0,25, 0,50, 1, 2, 10, 15, 30 et 45^o ;
- d'une convolution avec un filtre médian ;
- et d'une convolution avec un filtre gaussien.

Nous avons choisi ces variations car elle font partie des variations les plus utilisées dans le domaine et les plus faciles à appliquer. Elle permettent de modifier suffisamment le signal de l'image pour lessiver les marques cachées dans le signal de l'image par une méthode de tatouage d'images par exemple, ou de rendre leur détection impossible tout en préservant le contenu visuel ainsi que la qualité de l'image.

Chaque fragment a été utilisé pour interroger la base en suivant la même procédure décrite dans la section précédente. Une moyenne du taux de reconnaissance a été ensuite calculée sur l'ensemble des fragments de même proportion et ayant subi le même type de changement. Un image est reconnue si elle apparaît dans la liste des images semblables d'un fragment requête.

Les tableaux 5.(a), 5.(b),. . . ,5.(h) donnent les résultats obtenus pour une partie des expériences réalisées. Les résultats de la recherche des fragments ayant subi des rotations de 0,25, 0,50, 1, 15 et 30^o n'ont pas été présentés. Ils ont été, cependant, pris en compte lors du calcul des moyennes du taux de reconnaissance pour la totalité des variations (Tableau 5.(i))

Le tableau 5.(a) présente le taux de reconnaissance pour différentes stratégies de recherche lors de l'interrogation de la base en utilisant les fragments originaux (n'ayant subi aucune transformation). Il montre que le taux de reconnaissance est très élevé même lorsque la proportion de la taille du fragment par rapport à la taille de l'image originale est de 6,25%. Il montre aussi que le taux de reconnaissance n'est pas affecté par l'imprécision de la recherche de ppv.

Les tableaux 5.(b) à 5.(h) présentent les résultats obtenus pour la recherche de fragments ayant subi des variations. Naturellement, les taux de reconnaissance dans ces cas là sont inférieurs par rapport aux taux de reconnaissance obtenus pour les fragments n'ayant subi aucune transformation. Néanmoins, ils restent très bons dans le cas de variations de type, renforcement de contraste, cisaillement, convolution avec un filtre gaussien et convolution avec un filtre médian. Il sont, par contre, nettement moins bons pour les fragments ayant subi une rotation de plus de 30^o et plus particulièrement pour les petits fragments (6,25% et 9%). Dans ces deux derniers cas, nous remarquons que le taux de reconnaissance se dégrade considérablement lorsque le taux d'imprécision augmente. Ceci est dû principalement au petit nombre de descripteurs associés à ces fragments. En effet, seulement 4,4 (resp. 8,88) descripteurs en moyenne ont été calculés pour décrire le fragment dont la proportion est 6,25% (resp. 9%) dans le cas de la rotation à 45^o. Comme nous l'avons expliqué précédemment, plus le nombre de descripteurs de l'image requête est élevé, moins l'imprécision de la recherche de ppv a d'impact sur le résultat final.

Ce problème n'est cependant pas une vraie limitation du système, car, en pratique, soit la taille des fragments est trop petite et, par conséquent, le fragment n'a aucune utilité pratique, soit sa taille est suffisamment grande, auquel cas le nombre de descripteurs qui leurs sont associés sera élevé et le problème ne se pose donc pas.

Pour conclure cette expérience, nous avons calculé les taux de reconnaissance moyens pour l'ensemble des variations considérées (voir tableau 5.(i)). Ces résultats confirment les conclusions précédemment énoncées concernant la robustesse des descripteurs (des taux moyens de reconnaissance relativement élevés) et la capacité de la recherche de ppv approximative à réduire considérablement le temps de réponse par rapport à la recherche séquentielle de base tout en préservant la qualité des résultats finaux. Nous retrouvons aussi les problèmes de dégradation de la qualité des résultats lorsque les fragments sont trop petits.

6 Conclusion

Nous avons présenté dans cet article un schéma de recherche d'images par le contenu pour l'identification des copies. Il se base sur l'utilisation des invariants locaux différentiels pour la description des images et sur une méthode de recherche approximative de plus proches voisins pour accélérer les recherches. Nous avons montré expérimentalement la robustesse des descripteurs utilisés et leur capacité à absorber les variations usuelles que peuvent subir les images piratées ainsi que l'efficacité de la méthode de recherche approximative de plus proches voisins utilisée. Nous avons aussi évalué l'impact de l'imprécision de la recherche de plus proches voisins sur le résultat final tant sur le plan théorique que sur le plan expérimental.
Dans nos travaux futurs, nous envisageons d'élargir la campagne d'évaluation à d'autres types de variations et nous comptons adresser de très grandes bases d'images ($\approx$ un million d'images). Dans un tel contexte, de nouveaux problèmes de reconnaissance et de rapidité risquent d'apparaître.

Références

[1] K. S. Beyer, J. Goldstein, R. Ramakrishnan et U. Shaft. When is "nearest neighbor" meaningful? *Proceedings of the 7th International Conference on Database Theory, Jerusalem, Israel*, pages 217-235. Springer, January 1999.

[2] L. Amsaleg et P. Gros. Content-based retrieval using local descriptors: Problems and issues from a database perspective. *Pattern Analysis and Applications, Special Issue on Image Indexation*, 4:108-124, 2001.

[3] S.-A. Berrani, L. Amsaleg et P. Gros. Approximate searches: k-neighbors + precision. *Proceedings of the 12th* ACM *International Conference on Information and Knowledge Management, New Orleans, Louisiana, USA*, pages 24-31, November 2003.

[4] R. C. Veltkamp et M. Tanase. Content-based image retrieval systems: A survey. Technical Report UU-CS-2000-34, Department of Computing Science, Utrecht University, October 2000.

[5] L.M.T. Florack, B. ter Haar Romeny, J.J Koenderink et M.A. Viergever. General intensity transformation and differential invariants. *Journal of Mathematical Imaging and Vision*, 4(2):171-187, 1994.

[6] C. Schmid et R. Mohr. Local grayvalue invariants for image retrieval. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530-534, May 1997.

[7] Y. Dufournaud, C. Schmid et R. Horaud. Matching images with different resolutions. *Proceedings of the Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, USA*, volume 1, pages 612-618, June 2000.

[8] R. Deriche, V. Gouet et P. Montesinos. Differential invariants for color images. *Proceedings of the 14th International Conference on Pattern Recognition, Brisbane, Australia*, 1998.

[9] C. G. Harris et M. J. Stephens. A combined corner and edge detector. *Proceedings of the 4th Alvey Vision Conference, Manchester, England*, pages 147-151, September 1988.

[10] E.S. Page. Continous inspection schemes. *Biometrika*, 41:100-115, 1954.

[11] D.V. Hinkley. Inference about the change-point from cumulative sum tests. *Biometrika*, 58:509-523, 1971.

[12] M. Basseville. Detecting changes in signals and systems – A survey. *Automatica*, 24(3):309-326, 1988.

[13] S.-A. Berrani, L. Amsaleg et P. Gros. Recherche par similarité dans les bases de données multidimensionnelles : panorama des techniques d'indexation. *RSTI - Ingénierie des systèmes d'information. Numéro spécial « Bases de données et multimédia »*, 7(5-6):9-44, 2002.

[14] R. Weber, H.-J. Schek et S. Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. *Proceedings of the 24th International Conference on Very Large Data Bases, New York City, New York, USA*, pages 194-205, August 1998.

[15] P. Indyk et R. Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. *Proceedings of 13th Annual* ACM *Symposium on Theory of Computing, Dallas, Texas, USA*, pages 604-613, May 1998.

[16] K. P. Bennett, U. Fayyad et D. Geiger. Density-based indexing for approximate nearest-neighbor queries. *Proceedings of the 5th* ACM *International Conference on Knowledge Discovery and Data Mining, San Diego, California, USA*, pages 233-243, August 1999.

[17] S.-A. Berrani, L. Amsaleg et P. Gros. Recherche approximative de plus proches voisins : Application à la reconnaissance d'images par descripteurs locaux. *RSTI - Technique et Science Informatiques, numéro spécial « Indexation par le contenu visuel »*, 22(9):1201-1230, 2003.

[18] C.J. Van Rijsbergen. *Information Retrieval*. Butterworth, 1979.

[19] L. Amsaleg, P. Gros et S.-A. Berrani. Robust object recognition in images and the related database problems. *Special issue of the Journal of Multimedia Tools and Applications*, 2003. À paraître.