

Chapitre 1

TRAITEMENT DE SIGNAL MULTIMEDIA

Exemples de Traitements d'Images

Analyse d'images et de vidéos

Les quatre grands domaines d'application du traitement d'image ont été présentés de façon globale : quelle problématique ? quels objectifs ? avec quelles méthodes et quels outils ? Il est donc intéressant de s'attarder, à présent, sur des applications concrètes issues de ces différents domaines.

Analyse d'images et de vidéos

- Objectifs
 - Détection d'objets et extraction (segmentation)
Objets d'intérêt
 - Reconnaissance de formes
Classification et identification d'objets
 - Interprétation et analyse de la scène
 - Relationnelle
 - Description quantitative
 - Description qualitative

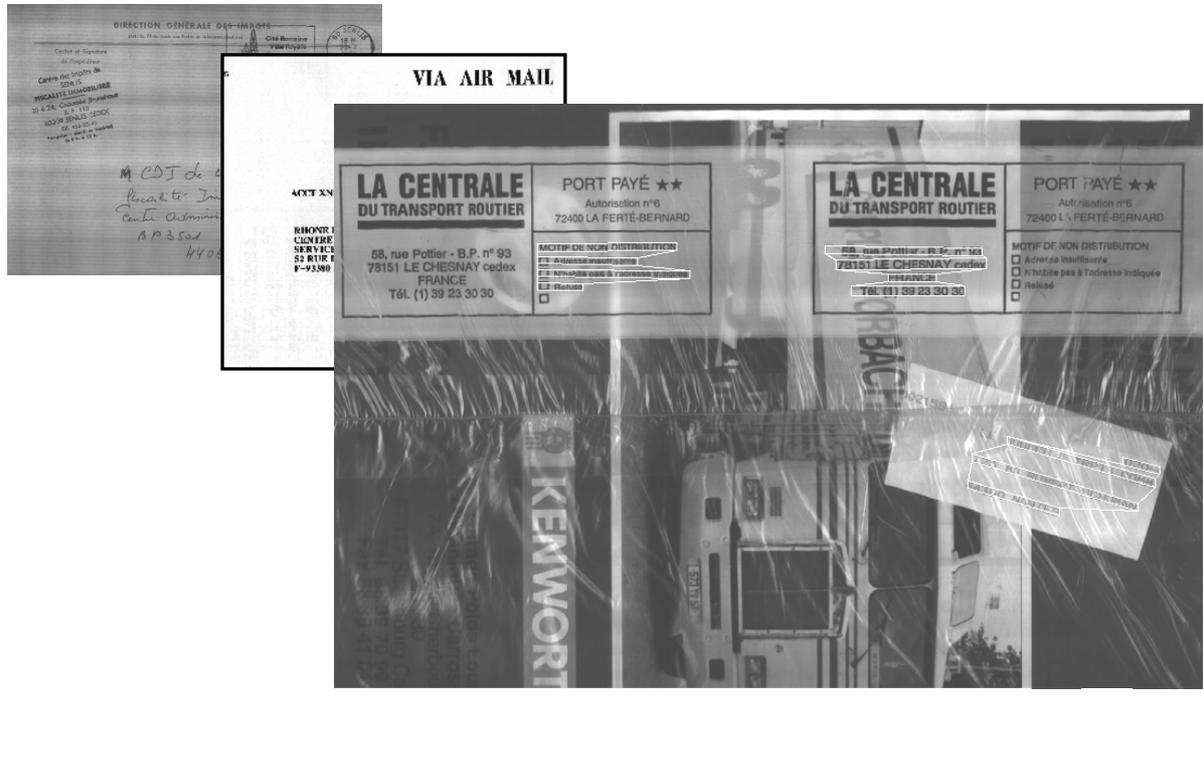
Le premier domaine auquel on s'intéresse est « l'analyse d'images ». Il recouvre un très grand nombre d'applications potentielles, c'est sans doute le domaine pour lequel les exemples sont les plus variés (analyse médicale : classification des chromosomes ; lecture automatique : authentification de signatures ; la télédétection : segmentation et classification de zones géographiques, ...).

On peut néanmoins proposer une démarche type en analyse d'images qui consiste à :

- **Détecter** les différents objets présents dans la scène et les extraire (segmentation, découpage en régions) ;
- **Caractériser** ces objets (dimensions, emplacement, ...) ;
- **Reconnaître** les objets et effectuer leur classification ;
- **Analyser** et **Interpréter** la scène en fonction des résultats précédents. La scène pourra, entre autres, être décrite de façon quantitative et/ou qualitative.

Il faut cependant rappeler que les applications sont très variées, et l'approche globale proposée ici peut parfois nécessiter quelques modifications dans les étapes de traitement, en fonction des difficultés rencontrées.

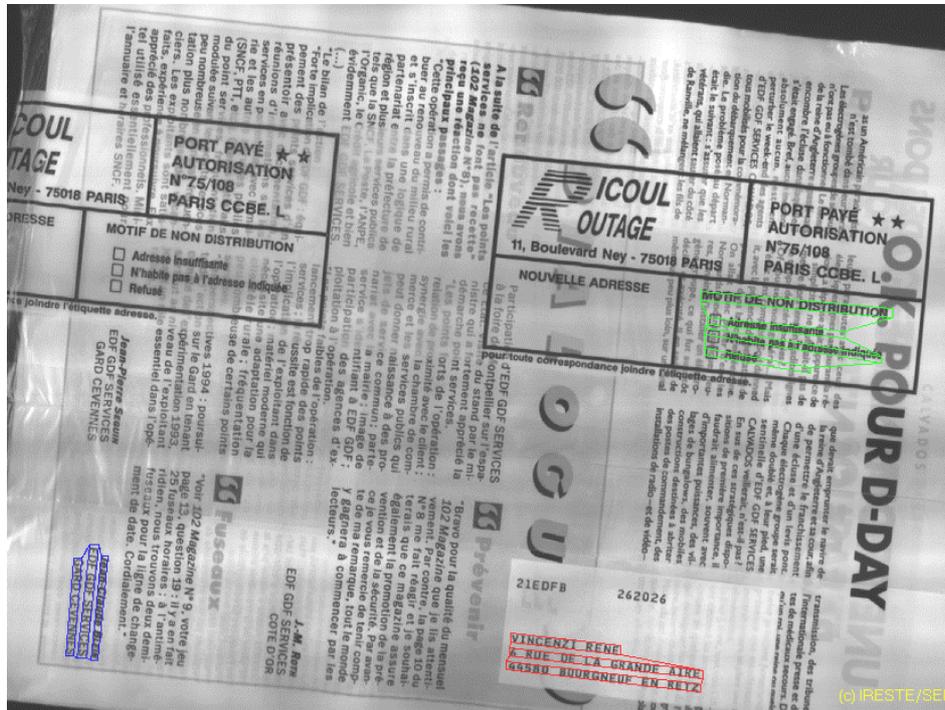
Détection et extraction d'un bloc adresse



L'exemple, présenté sur la figure ci-dessus, met en avant la détection et l'extraction d'une adresse de livraison. L'application servirait au tri automatique dont la nature dépend des objets postaux et des modes de dépôts. La plupart des pays industrialisés utilisent des systèmes de tri automatique, les obstacles varient d'un pays à l'autre en raison des différences qui existent, entre autres, pour les codes postaux (chiffres exclusivement ? taille ? ...).

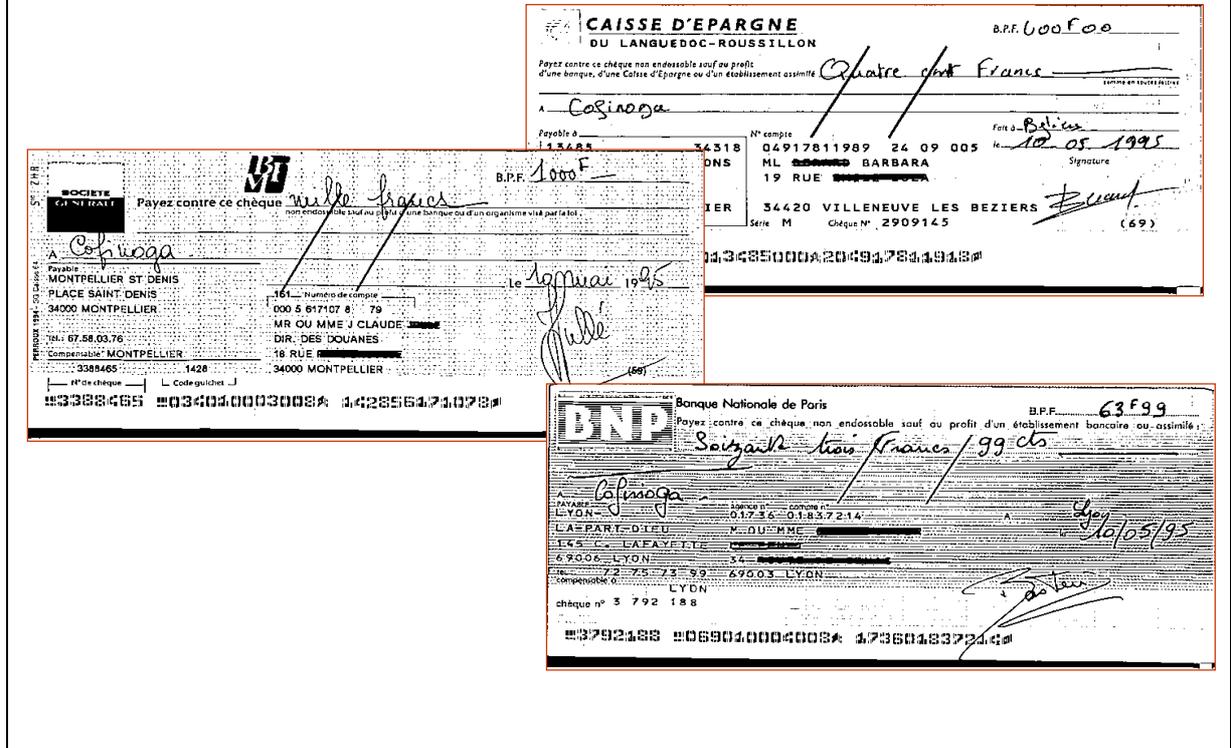
Divers algorithmes de traitement existent pour y parvenir. On peut par exemple remarquer que les caractères du texte forment une texture régulière contenant des contours verticaux allongés horizontalement. En appliquant des filtres directionnels, il est alors possible de mettre en avant les pixels de l'image liés au texte. On peut par exemple effectuer une binarisation suivie d'un post traitement afin d'extraire les rectangles englobant des zones de texte. Le texte étant séparé du reste de l'image, on pourrait ensuite effectuer une analyse des composantes connexes pour trouver celles correspondant aux zones du texte. Le traitement global de l'image étant effectué, il faudra alors appliquer des traitements locaux et considérer des contraintes diverses (par exemple géométriques) pour éviter les erreurs. Ces différents traitements seront présentés dans les chapitres à venir.

Détection et extraction d'un bloc adresse



L'image ci-dessus est également le résultat d'un traitement d'analyse. L'objectif est de nouveau de détecter et d'extraire sur l'image les zones de textes liées à l'adresse du destinataire. Cependant l'image ci-dessus présente de nouvelles difficultés : l'arrière plan semble beaucoup moins uniforme que celui présenté précédemment, car il est composé de texte. Il faut donc également être en mesure de distinguer parmi les caractères qui se chevauchent, ceux qui sont liés au « bloc adresse de l'expéditeur » et ceux qui sont liés au reste de l'image. Cependant le texte de l'arrière plan et le texte lié à l'adresse sont perpendiculaires l'un à l'autre. Il paraît donc possible de segmenter l'image en appliquant des filtres directionnels adaptés. Puis reprendre la chaîne Caractérisation – Reconnaissance – Interprétation d'un traitement d'analyse d'images, afin d'obtenir comme sur l'image présentée une détection précise du bloc d'adresse destinataire.

Extraction et lecture du montant sur les chèques



Le dernier exemple présenté pour l'analyse d'image concerne l'extraction et la lecture du montant inscrit sur un chèque. Elle bénéficie d'une attention particulière compte tenu de ses enjeux industriels et économiques. De nombreuses publications et méthodes existent dans le domaine, les utilisateurs potentiels de la reconnaissance automatique semblent néanmoins en attente de résultats plus fiables. De ce fait, le problème est considéré comme étant encore ouvert. Il s'agit donc principalement d'une reconnaissance de chiffres et de lettres manuscrites dans un environnement dont la complexité impose l'utilisation de techniques appropriées. En effet, outre la grande variabilité dans la morphologie des caractères d'un individu à l'autre, il est nécessaire de prendre en considération des problèmes tels que la non standardisation du format d'un chèque (position des champs, type de fond) et le chevauchement des entités composant le montant (chiffres, ligne de base, séparateur décimal).

Les principaux produits ou prototypes disponibles en France ont pour origine : le SRTP (La Poste) avec Dassault AT (Automate de remise de chèques lisant le montant numérique seulement) ; Matra (MSI) et son produit LireChèques ; et enfin le produit InterCheque de a2ia qu'on retrouve chez plusieurs banques (Société Générale, Banque Populaire, ...) et dont les taux de reconnaissance avoisinent les 70 % grâce à une technologie s'appuyant sur les approches par réseaux neuronaux et sur les modélisations Markoviennes.

Exemples de Traitement d'images

Codage des images et vidéos avec compression d'information

Après ces quelques exemples d'applications de l'analyse d'images, nous allons à présent nous intéresser aux applications liées au codage d'image avec compression de données.

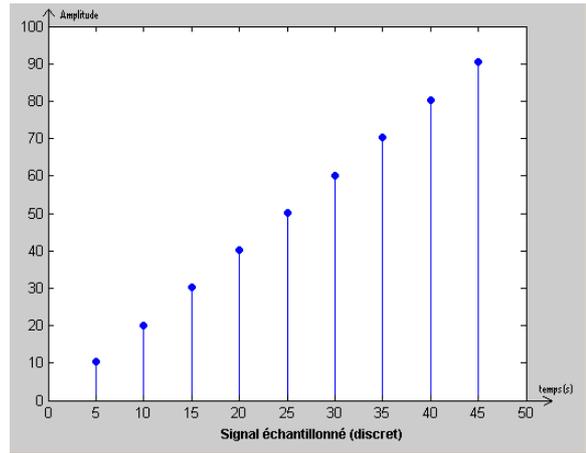
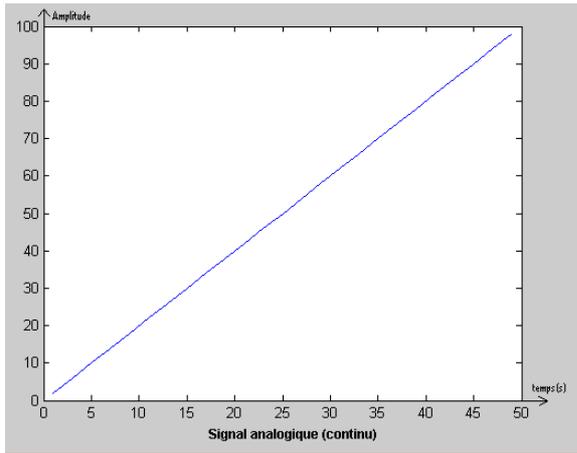
Contexte

- Extension des techniques numériques à plusieurs domaines du traitement d'information : capture , stockage et transmission
 - ➡ désir d'une méthode unique de stockage, traitement et transmission indépendamment de l'image
- Les signaux audio-visuels ont des propriétés qui requièrent des méthodes dédiées à une représentation efficace de ces derniers :
 - les données sont la représentation d'un signal analogique
 - ➡ échantillonnage + quantification
 - grande quantité de données pour représenter une petite partie de la source d'information originale (ex: 1s d'audio/vidéo)

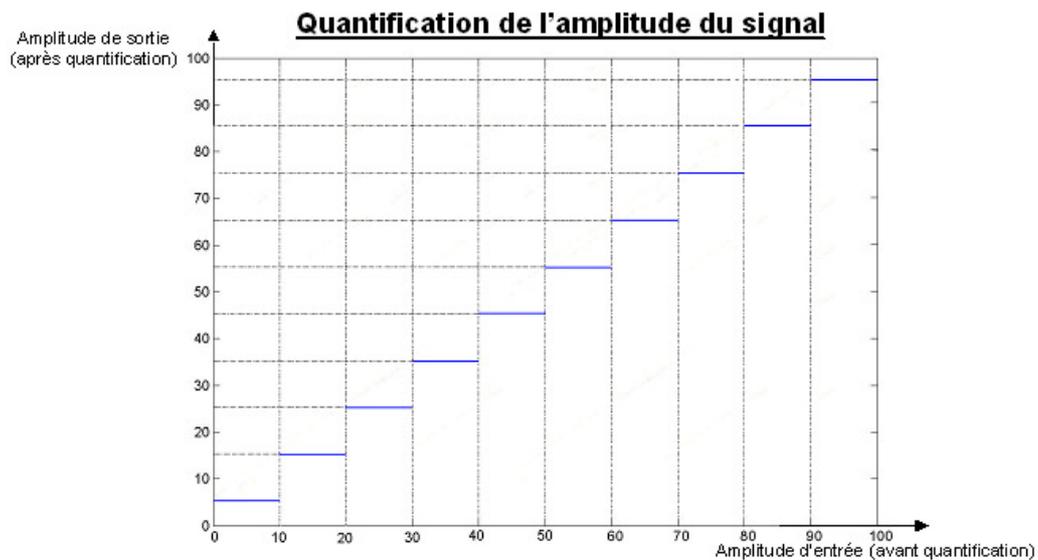
Comme cela a été indiqué dans la présentation, la problématique résulte de la nécessité de réduire de manière importante la quantité d'éléments binaires d'information nécessaire à la représentation des images à des fins de mémorisation ou de transmission à distance. Bien que les images soient très variées, et qu'en conséquence les informations qu'elles contiennent puissent toutes être très différentes (couleur, lumière, objets, ...), il demeure préférable de mettre en place une procédure unique de codage à compression d'information, que l'on appliquera à chaque image indépendamment de ses caractéristiques.

Par ailleurs, les signaux audiovisuels sont des signaux analogiques (continus) dont la représentation nécessite une très importante quantité de données (ex : pour une seule seconde de vidéo, il faut être capable de stocker une séquence de 25 images). La représentation efficace de tels signaux requière donc la mise en place de méthodes adéquates.

La représentation binaire d'un signal analogique n'est possible que si ce dernier a subi un échantillonnage (classiquement temporel, spatial, ...), afin de pouvoir travailler sur un nombre fini d'échantillons de ce signal. Les amplitudes liées à un signal analogique s'étendent également sur un intervalle continu qu'il va falloir discrétiser selon différents critères : il s'agit de la quantification du signal. Ces deux étapes sont présentées ici brièvement pour un signal 1-D, mais elle seront reprises et détaillées dans la suite de ce chapitre pour un signal image 2-D.



La figure ci-dessus représente un signal de type rampe avant et après échantillonnage. La fréquence d'échantillonnage est égale à $0,2\text{Hz}$ ($\frac{1}{5}\text{s}^{-1}$). L'amplitude de ce signal varie continûment sur l'intervalle $[0, 100]$, il faut donc effectuer une quantification.

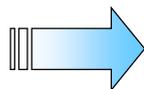


La figure ci-dessus représente un exemple de quantification : si un échantillon du signal a une amplitude comprise dans l'intervalle $[10k ; 10(k + 1)[$, « k » étant un entier naturel, sa valeur d'amplitude sera ramenée par quantification à $10k + 5$.

Après quantification, les données brutes représentent une grandeur discrète qu'il va être possible de coder. Typiquement on utilise un codage binaire.

Grande quantité de données pour représenter les informations visuelles

- Conditions sur les tailles de données
 - une page Télétex = 25 lignes × 40 char ⇒ 1 Kbyte
 - une page A4 de texte = 50 lignes × 80 char ⇒ 4 Kbytes
 - une seconde de téléphone (300 Hz - 3400 Hz) ⇒ 8 Kbytes
 - une seconde d'audio en hi-fi stéréo ⇒ 176.4 Kbytes
 - un fax A4 ⇒ 1.1 Mbytes
 - une image couleur (qualité diapositive: 2000 × 3000) ⇒ 18 Mbytes
 - une seconde de programme TV (signal vidéo) ⇒ 21 Mbytes
 - une seconde de programme HDTV ⇒ 230 Mbytes



Besoin de compression de données

Pour illustrer les problèmes liés à la quantité nécessaire de données pour représenter peu d'information d'un signal image, le document ci-dessus donne à titre indicatif les tailles de données requises pour stocker et transmettre différentes images, que l'on trouve régulièrement dans notre environnement. Attention, les tailles sont proposées ici en unités anglo-saxonnes : 1 byte = 1 octet = 8 bits.

Par exemple, une simple feuille standard (A4) de fax nécessiterait pour la transmission de l'image de base une capacité de 1,1Mo. Ces différents exemples montrent combien il est important de réaliser un codage à compression de données lors d'une transmission ou d'un stockage d'image.

Contexte

- Besoin de mélanger dans le même document différents types d'informations
 - Texte (commun ; intrinsèquement structuré ; niveau sémantique)
 - Tables de données (communes ; intrinsèquement structurées)
 - Graphiques (données vectorielles ; information structurée)
 - Audio (liste de nombreux échantillons temporels ; faible niveau de représentation (non structuré et non sémantique))
 - Image (liste d'échantillons scalaires/vectoriels issus d'un scannage plan 2-D ; faible niveau de représentation (non structuré et non sémantique))
 - Vidéo (groupe de données « Image » selon l'axe du temps ; même propriétés que la donnée « Image »)

Outre le fait qu'une grande quantité de données soit nécessaire au stockage et à la transmission de peu d'information dans le cas d'un signal image, la représentation de ce signal est également rendue difficile par la nature même du signal image. En effet, ce signal mélange des types d'informations parfois nombreux et variés. De fait, une image peut comporter des zones de textes, des tables de données, des graphiques, ...

Les compromis « Taux de compression/Qualité » dépendront donc aussi directement du contenu de l'image. Si on considère par exemple une carte géographique, accompagnée d'une légende, il ne serait pas possible de se contenter de pouvoir reconnaître les formes, et les frontières des divers continents, mers, océans, ... après décodage. L'écriture ne devra pas être trop dégradée, afin que légendes et noms figurant sur la carte restent facilement lisibles pour tout utilisateur.

Principaux aspects de la compression d'images

- Type d'images à compresser ? Pour quels services ?
- Qualité de l'image :
 - codage d'image **sans perte**
 - ou codage d'image **avec perte**
dans ce cas : avec ou sans dégradations visibles?
- **Rapport Taux de compression / Qualité**
- Complexité : codage , décodage
- Sensibilité aux erreurs (*robustesse face aux erreurs*):
 - effets des erreurs de transmission sur l'image décodée
- Taux de compression fixe ou variable :
⇒ conditions de transmission

Avant d'appliquer à une image un codage à compression de données, il faut se poser quelques questions, parfois triviales, mais qui influent généralement fortement sur les résultats obtenus après la compression :

- À quel type d'image a-t-on affaire ? pour quelle utilisation ?
- Faut-il un codage exact (réversible, mais forte contrainte et donc faible valeur du taux de compression) ? sinon les dégradations peuvent-elles être ou non visibles par l'œil humain (là aussi, le taux de compression sera directement affecté) ?
- Le taux de compression doit être d'autant plus fort que l'on veut gagner en capacité de stockage et de transmission d'images, mais la qualité en sera affectée, quel équilibre souhaite-t-on alors ?
- Une augmentation du taux de compression entraîne une hausse de la complexité du décodeur et donc de la charge de calculs pour le système. Là aussi, il faut trouver un juste équilibre en fonction de l'utilisation prévue.
- Quelle robustesse face aux erreurs ? car les erreurs de transmission pourront avoir des répercussions directes fortes sur l'image décodée.
- Le taux de compression est-il fixe ou variable ? dans quelles conditions de transmission se trouvera-t-on ?

En définitive, l'utilisation que l'on veut faire du codage est aussi importante que le codage lui-même. Il faut donc définir avant de commencer le type de compression, et en fonction de ses propres objectifs, choisir le rapport taux de compression/qualité que l'on souhaite obtenir.

En résumé

- **Applications complexes**
 - important volume d'information
 - large champ d'applications et de services

photographie numérique (MMDB)



HDTV

- **Dépendances multiples**
 - quelles images (\approx types et services)
 - quels taux d'images/s, résolution,....
 - quelle qualité requise (*niveau d'acceptabilité*)
 - quel réseau de communication
- **Conditions multiples de résolutions**
(multi-résolution, qualités ...)
- **Autres fonctionnalités que le codage**

En résumé, ce qui vient d'être présenté conduit à la conclusion que la mise en place d'un codage d'image à compression de données est une procédure complexe. D'une part, un très important volume d'information et un large spectre d'applications et de services sont considérés. D'autre part, le codage en lui-même dépendra de nombreux paramètres : l'image à traiter (type, contenu, utilisation, ...); la résolution; la qualité nécessaire à l'utilisation envisagée et le seuil de dégradation accepté; le réseau de transmission utilisé... À ces difficultés peut venir s'ajouter une complexité, liée à des fonctionnalités autres que le codage lui-même.

La suite présente le cas d'un codage pour une image haute définition, cette dernière génère, de par sa nature, une contrainte qui influe directement sur la complexité de la compression.

Exemple d'une image HD : Image *Bike*

(décodée à trois niveaux résolution depuis le même traitement de codage)



La figure ci-dessus présente l'image haute définition (HD) « *Bike* » qui après codage à compression de données, a été décodée à 3 niveaux de résolution et de tailles différents, mais en conservant le même traitement. La présentation est faite de façon pyramidale, de la résolution la plus grossière (à gauche de l'image) à la plus fine (à droite). L'image qui possède la plus grande résolution apparaît nette, les couleurs et les détails sont bien restitués. Les dégradations ne semblent pas perceptibles par le système visuel humain dans ce cas. Cependant, pour une image HD telle que « *Bike* » à la résolution 2048×2560 , de nouveaux problèmes de codage à compression apparaissent.

Remarque : attention toutefois à la représentation donnée sur la figure ci-dessus, si l'affichage ne permet pas de garder les mêmes résolutions que celles d'origine, il y a un risque de sous-échantillonnage.

Problèmes

- *Numérisation des signaux audio-visuels*
 - problèmes spécifiques avec les images / vidéo HD
 - capteurs d'image (capteurs multi-spectral, résolution, sensibilité, capture à grande vitesse de luminance)
 - quantification (grande précision pour l'audio / quelques systèmes d'imagerie, grande vitesse de conversion)
- *Besoin d'un fort taux de compression en gardant une bonne qualité*

Pour stockage et transmission (réduction de bande passante)
- *Autres conditions*
 - adaptations multiples (taux de transmission, résolution, robustesse face aux erreurs, paquets)
 - accès au contenu partiel, description du contenu, ...



Standards JPEG2000, MPEG4, MPEG7

Lors de la numérisation des signaux audiovisuels, des problèmes spécifiques aux images « haute définition » (HD) apparaissent :

- Le type de capteur à employer (capteurs de type multispectral, résolution, sensibilité, capture à grande vitesse de luminance) ;
- La manière dont est choisie la loi de quantification.

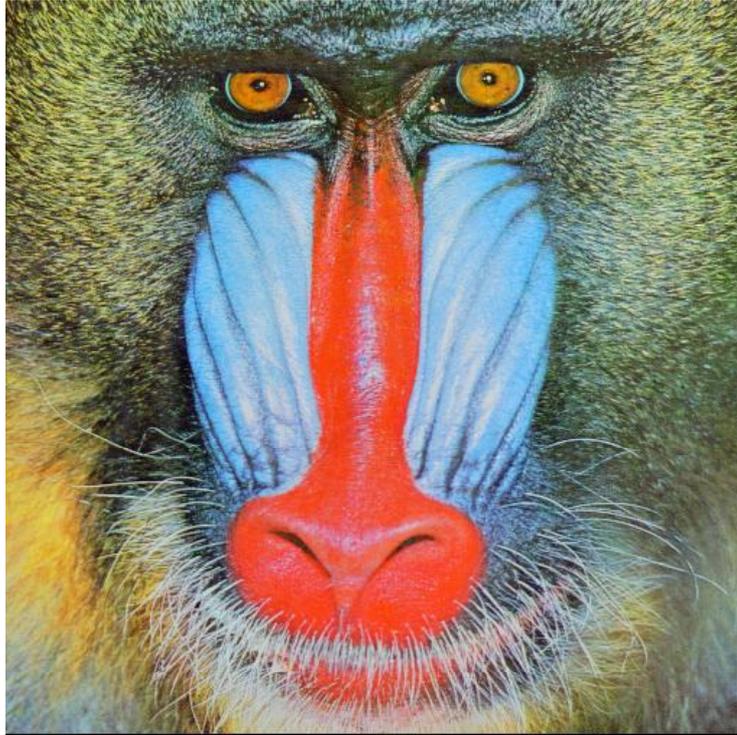
De plus, on cherche à atteindre des taux de compression importants, au vue des applications envisagées. Cependant, par définition, on souhaite conserver un degré élevé de qualité (image haute définition). Le compromis entre le taux de compression et la qualité atteint donc des limites. Enfin, en prenant en compte d'autres types de contraintes, tels que la nécessité de l'adaptation du codage (fonction de la robustesse aux erreurs, du mode de transmission, ...), il est apparu essentiel de créer des standards de compression qui pourraient répondre aux différentes contraintes dégagées par ces problèmes et s'appliquer aux images indépendamment de leur nature. Les standards les plus répandus pour le codage et la compression des images et de la vidéo sont : JPEG, GIF, JPEG2000, et MPEG4.

La suite présente, à titre d'illustration, différents résultats obtenus pour l'image « *Mandrill* » qui a été codée en JPEG avec des paramètres et des taux de compression différents.

Exemple de codage JPEG : original de *Mandrill*

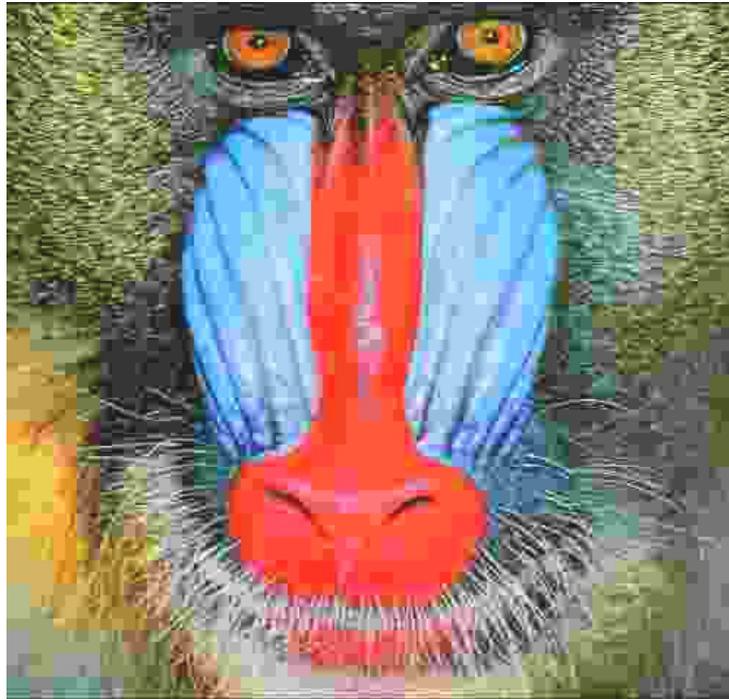
Original:
512 × 512
⇒ 24 bpp

⇒ 768 Ko



Ci-dessus l'image « *Mandrill* » originale avec une taille de 512×512 pixels. Un pixel est codé sur 24 bits (trois fois 8 bits car la représentation initiale de l'image couleur est R-V-B, il y'a donc 2^{24} niveaux possibles pour chaque pixel). L'image est donc représentée par un volume de $512 \times 512 \times 24 = 768$ Ko. Nous allons faire subir à cette image une série de codages JPEG, avec différents paramètres, pour comparer la qualité visuelle des résultats et les taux de compression obtenus.

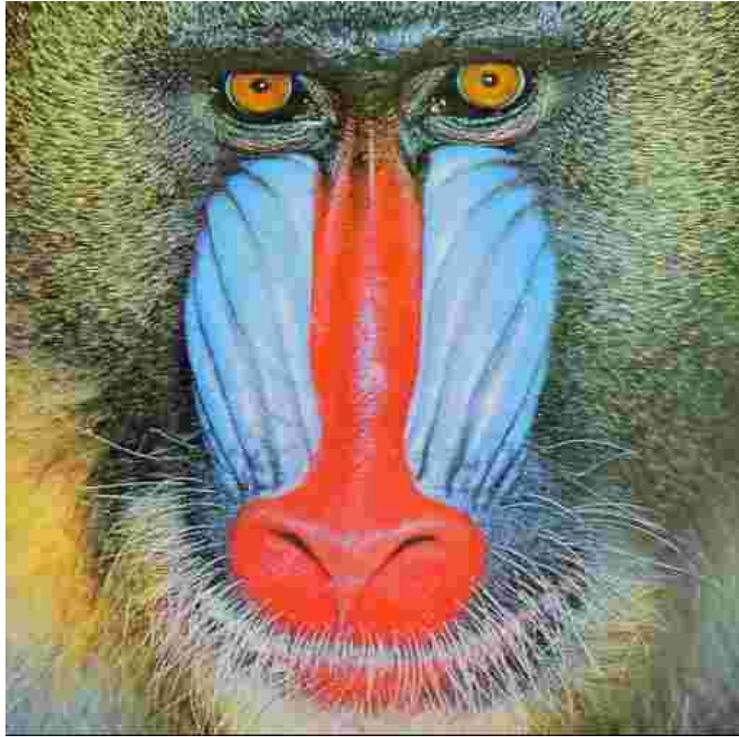
Exemple de codage JPEG (q = 5 ; 9 Ko)



$$\tau_C = 768/9 \\ \approx 85$$

Ci-dessus l'image « *Mandrill* » qui a subi un codage JPEG. La quantification a été réalisée en utilisant un facteur de qualité q faible ($q = 5$). Le gain en compression est donc très important. La taille des données décrivant l'image n'est plus que de 9Ko. Le taux de compression τ_C est très fort, et vaut $768/9 = 85$. Cependant, l'image a énormément perdu en qualité visuelle. Les formes et les couleurs sont beaucoup plus grossières que sur l'image originale, et un effet de bloc est apparu très nettement sur l'ensemble de l'image (des blocs de pixels sont représentés quasiment par la même couleur).

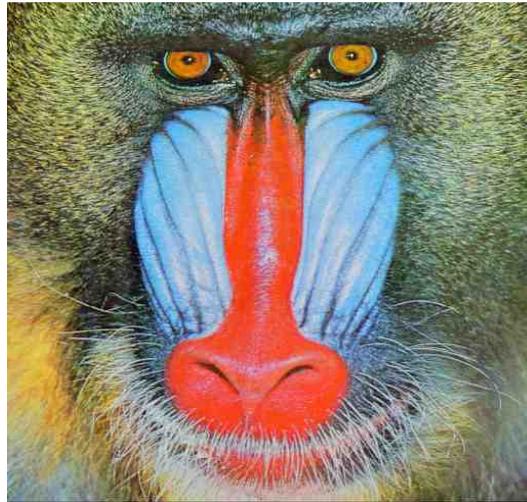
Exemple de codage JPEG (q = 10 ; 17 Ko)



$$\tau_C = 768/17 \\ \approx 45$$

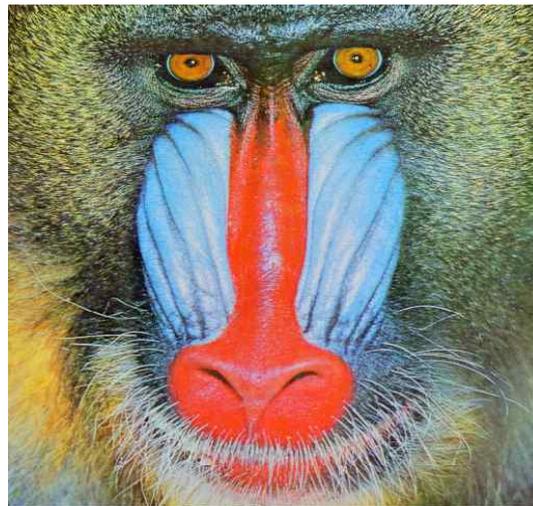
Un autre exemple de résultat avec un facteur de qualité plus fort quoique restant faible (q = 10). Le taux de compression est moins important que précédemment ($\tau_C = 45$). L'image a bien gagné en qualité, et l'effet de bloc est beaucoup moins apparent, bien qu'il soit encore visible sur les couleurs vives rouges et bleues du museau du babouin. La gamme de couleurs est moins importante que sur l'image originale, mais elle permet tout de même d'obtenir un meilleur rendu sur le pelage du singe, que lors du codage précédent. Notons qu'en doublant le facteur de qualité q, le taux de compression a été divisé par 1,88 : il est presque deux fois moindre.

Exemple de codage JPEG (q = 25 ; 32 Ko)



$$\tau_c = 768/32 \\ \approx 24$$

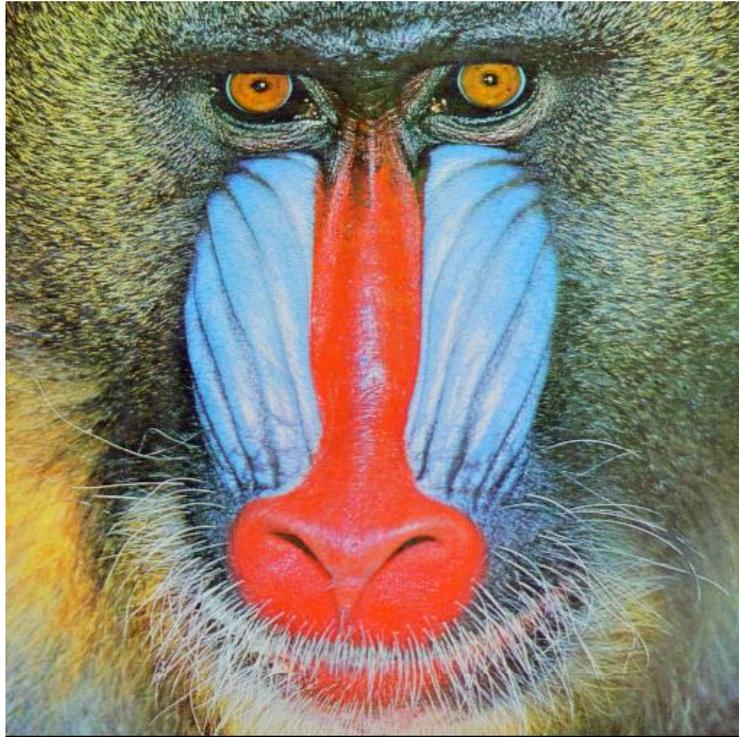
Exemple de codage JPEG (q = 35 ; 40 Ko)



$$\tau_c = 768/40 \\ \approx 19.2$$

Deux autres résultats obtenus en cherchant à privilégier encore un peu plus la qualité de l'image obtenue par rapport au gain en compression. Cependant le taux de compression n'est plus divisé que par 1,25 cette fois-ci. L'évolution du taux de compression en fonction de la quantification n'est en général pas une fonction linéaire.

Exemple de codage JPEG (q = 75 ; 76 Ko)



$$\tau_C = 768/76 \\ \approx 10.1$$

Enfin l'image ci-dessus présente un résultat où la qualité a été fortement privilégiée par rapport à l'importance de la compression. Le nombre de bits nécessaires au stockage de l'image codée est simplement 10 fois moins important que celui nécessaire au stockage de l'image originale. Il est cependant très difficile pour le système visuel humain de détecter les dégradations subies par cette image : la différencier de l'image d'origine au niveau de la gamme de couleur et de la netteté des contours et des textures n'est guère possible.

Le traitement d'image a donc été présenté sous différents aspects. Il est né des nouveaux besoins qu'ont engendrés les nouvelles technologies, et s'est grandement appuyé sur ces mêmes nouvelles technologies pour se développer très rapidement. Avec son expansion se sont dégagés quatre grands domaines d'application : la restauration, le codage, l'analyse, et la synthèse d'images.

Dans ce cours, les domaines de l'analyse et du codage ont ensuite été accompagnés d'exemples d'applications concrètes, afin d'obtenir une première vision des possibilités et des résultats qu'offrent les différents traitements d'image.

Toutefois, dans la nature, les images qui nous entourent ne peuvent généralement pas être traitées directement, et ne sont pas numériques. Il faut donc les numériser pour pouvoir les traiter avec des algorithmes et des systèmes adaptés. C'est à cette étape de numérisation que nous nous intéresserons dans la ressource suivante.