pratique, il ne reste pas longtemps secret. On suppose ici que chaque clé est propre à un ayant droit, clé qui permet de l'identifier. Parallèlement, on met au point un système de gestion des clés, reposant sur un «tiers de confiance». Ce dernier est une entité sécurisée et angélique qui ne triche ni ne ment jamais : c'est elle qui va gérer les clés et opérer les vérifications. Cette opération délicate ne peut être confiée, ni à l'ayant droit présumé du document, car il pourrait mentir, ni à quelqu'un d'autre, car il faut la clé pour opérer cette vérification, et si celle-ci est révélée au grand jour, les marques qu'elle a produites deviennent très vulnérables.

Revenons aux techniques d'incrustation proprement dites. Le tatouage est engendré par des techniques de traitement du signal. Nous envisagerons ici le tatouage des images fixes (les contraintes du tatouage de la vidéo sont légèrement différentes : une image vidéo étant de moins bonne qualité, il est plus facile d'y camoufler une marque, mais, d'un autre côté, l'algorithme de tatouage doit être très rapide).

L'évolution des algorithmes de tatouage proposés depuis cinq ans est très nette : on est passé de solutions simples, naïves et peu résistantes à des techniques complexes et plus robustes. Ces dernières reposent sur la modification des coefficients de la transformée de Fourier (bidimensionnelle) de l'image *(voir la figure 3)*. Ces approches engendrent des filigranes de bonne résistance. La détection du filigrane repose en général sur la présence on non d'une structure particulière dans l'image ; on cache par exemple un signal présentant des propriétés de corrélation très spécifiques *(voir la figure 4)*. La conception d'un algorithme de tatouage/vérification commence donc par le choix de la structure de modification de l'image ; ce choix est contraint par la spécificité de l'utilisation (degré d'invisibilité, robustesse, rapidité d'exécution, mémoire utilisée).

Le tatouage impose un certain nombre de compromis entre la quantité d'information cachée, l'invisibilité et la robustesse de la marque. Il faut également tenir compte, lors de la conception de l'algorithme de vérification, des risques de non-détection (l'ayant droit présumé est réellement le propriétaire du document, mais l'algorithme n'arrive pas à le prouver) et de fausse alarme (on identifie une personne comme étant l'ayant droit du document, alors que c'est faux). Il convient aussi de définir si la présence de l'image originale est nécessaire à la vérification.

Les techniques d'incrustation se répartissent en deux catégories : les «spatiales» qui modifient directement les pixels (un pixel est un élément d'image), et les «fréquentielles» qui changent les coefficients des transformées de Fourier. Les algorithmes présentés ci-après donnent un panorama des techniques employées et des concepts qui entrent en jeu. Ces algorithmes opèrent uniquement sur la luminance (intensité lumineuse par unité de surface) des pixels *(voir la figure 2)*.

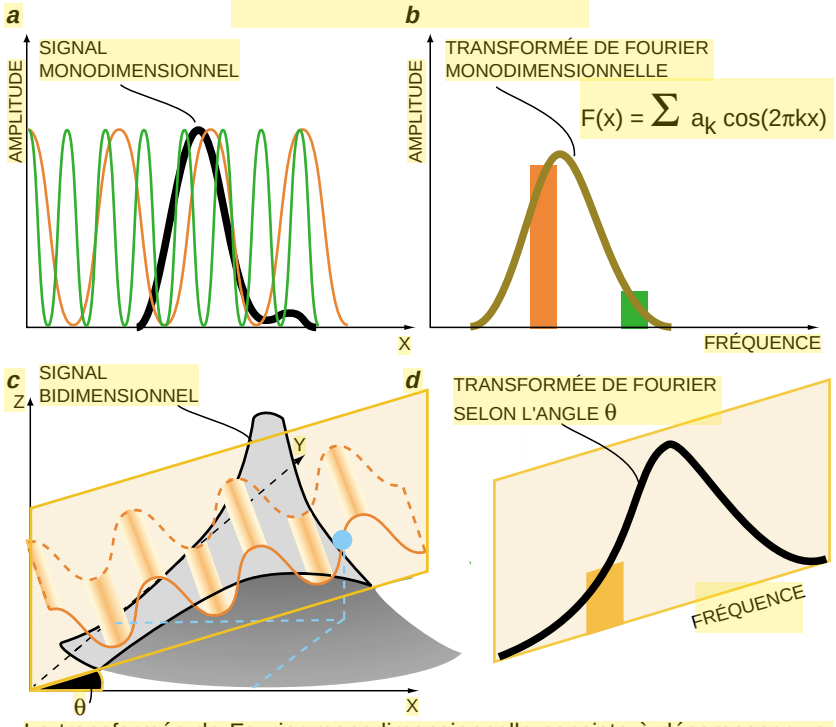
LA PROGRESSION DES TECHNIQUES

La première technique proposée est simple : une clé secrète détermine, dans une zone de l'image, l'emplacement de pixels. La luminance de chacun d'entre eux est codée par une valeur comprise entre 0 (en binaire 00000000), et 255 (en binaire 11111111). Le bit de poids faible de l'écriture binaire, le dernier, est celui qui a le moins d'influence sur la valeur de la luminance. Statistiquement ces bits de poids faible ont une probabilité 1/2 de valoir 0, et 1/2 de valoir 1. On cache l'information en introduisant un biais dans cette proportion : pour ce faire, on impose la valeur 1 aux bits de faible poids de certains pixels, choisis selon la clé. Cette modification sera imperceptible pour l'œil car la luminance des ces pixels aura varié d'au plus 1. On vérifie alors, sur l'image modifiée, la présence ou non d'un tatouage, en analysant les bits de poids faible correspondant aux pixels associés à la clé : si ces bits sont égaux à 1, alors l'image a été tatouée avec la clé (et elle appartient à l'ayant droit associé à cette dernière) sinon, elle n'a pas été tatouée avec cette clé (et n'appartient pas à cet ayant droit).

Malheureusement, les bits de poids faible des octets sont modifiés quand on applique la moindre transformation à l'image, en particulier lors d'une compression avec JPEG.

En 1995, W. Bender et ses collègues de l'Institut de technologie du Massachusetts ont proposé un algorithme dénommé *Patchwork,* qui repose sur la différence de luminance observée entre deux ensembles de pixels. Il est fondé sur l'analyse statistique suivante : si l'on considère deux grands ensembles *A* et *B* de

3. LA TRANSFORMÉE DE FOURIER



La transformée de Fourier monodimensionnelle consiste à décomposer un signal ne dépendant que d'un seul paramètre en une superposition de fonctions sinusoïdales de différentes fréquences *(a)*. À chacune de ces fréquences correspond un coefficient (a_k) de la transformée, dont la valeur mesure la contribution de cette fonction sinusoïdale *(b)*. De la même manière, on considère la transformée de Fourier d'un signal bidimensionnel, dépendant de deux paramètres *(c)*. Dans une direction θ, on décompose le signal selon des ondulations de différentes fréquences. Les contributions des différentes fréquences sont les coefficients de la transformée de Fourier dans cette direction.