Projet Image - Compte-rendu 2

Sécurité Visuelle - Obscuration d'image

Loïc Kerbaul - Valentin Noyé

28 octobre 2024

1 Introduction

Nous discutons dans ce compte rendu des 4 méthodes classiques de chiffrement de l'image que nous avons implémentées (floutage, masquage, pixélisation, chiffrement AES), ainsi que leur évaluation visuelle et par divers outils (PSNR, SSIM, entropie).

2 Implémentation

2.1 Floutage

Nous avons utilisé un filtre moyenneur pour implémenter notre méthode de floutage d'images. La taille du filtre utilisé est donnée au moment de l'exécution du programme (celle-ci doit être impaire). Ce filtre floute donc les pixels de la zone sélectionnée en fonction de ses voisins. Dans le cas où le voisin recherché n'existe pas (s'il se trouve en dehors des limites de l'image), alors le nombre de pixel pris en compte dans le calcul de la moyenne est évidemment adapté en conséquence.

Utilisation :

```
./floutage <entrée> <sortie> <x1> <y1> <x2> <y2> <taille du filtre>
```

(a) Image originale (b) Filtre de taille 3 (c) Filtre de taille 15 (d) Filtre de taille 31

FIGURE 1 – Floutage de la région

Sur les résultats ci-dessus, on observe que plus la taille du filtre est grande, plus l'effet du floutage est important.

2.2 Masquage

La technique de masquage consiste simplement a remplacer la région sélectionnée par un autre contenu. Dans notre cas, on utilise une couleur unie (dont les intensités de rouge de vert et de bleu sont données via les paramètres `R`, `G` et `B` à l'exécution du programme).

Utilisation :

```
./masquage <entrée> <sortie> <x1> <y1> <x2> <y2> <R> <G> <B>
```

(a) Image originale (b) Masque de couleur (0, 0, 0) (c) Masque de couleur (255, 255, 255) (d) Masque de couleur (212, 129, 118)

FIGURE 2 – Masquage de la région

2.3 Pixélisation

On effectue une pixélisation de la région sélectionnée en moyennant les pixels contenus dans chacun des «gros pixels», dont la taille est donnée en paramètre au programme.

Utilisation :

```
./pixelisation <entrée> <sortie> <x1> <y1> <x2> <y2> <taille de pixel>
```

(a) Image originale (b) Taille de pixels 5 (c) Taille de pixels 10 (d) Taille de pixels 20

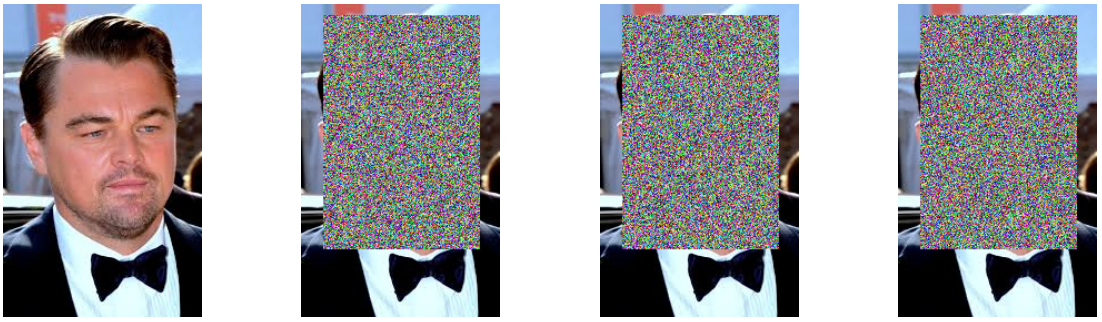
FIGURE 3 – Pixélisation de la région

2.4 Chiffrement

Pour l'obscuration par chiffrement, la librairie PlusAES est utilisée, et permet de chiffrer la région sélectionnée en mode CBC à partir d'une clé sur 128 ou 256 bits. Cela présente une approche réversible à l'obscuration d'une image.

Utilisation :

```
./aes <entrée> <sortie> <x1> <y1> <x2> <y2> <clé de chiffrement>
```



(a) Image originale (b) `abcdef0123456789` (c) `0123456789abcdef` (d) `EncryptionKey128`

FIGURE 4 – Chiffrement par AES en mode CBC de la région

3 Évaluation des méthodes

Nous constatons que les diverses méthodes implémentées présentent un ou plusieurs enjeux quant à leur fidélité visuelle et leur sécurité, ce qui pose ici un compromis important. En effet, nous sommes capables d'en tirer beaucoup d'informations sur l'image originale lorsque celle-ci est obscurcie par floutage ou pixélisation. En agrandissant la taille du noyau de floutage ou encore la taille des pixels lors de la pixélisation, nous nous rapprochons de ce qui constitue un masquage de la région. Et un tel masquage introduit beaucoup de distorsion par rapport à notre image originale, ce qui impacte la qualité visuelle de nos résultats. Le chiffrement, quant à lui, permet à notre image d'être obscurcie de manière réversible, ce qui est parfois nécessaire, mais le bloc chiffré, notamment avec AES dans notre cas, détonne totalement l'image résultante.

Afin d'interpréter ces résultats autrement que par une approche visuelle, nous introduisons ici le PSNR et le SSIM qui nous est familier. Toutes ces évaluations ont été menées sur la même image, à savoir le visage de Leonardo DiCaprio (Image source tirée de Wikipédia), ainsi que sur la même région rectangulaire. Nous pouvons également corréler nos résultats avec l'entropie obtenue pour chaque technique d'obscuration de l'image, à savoir que l'image originale a une entropie de 7,19358 bits/canal/pixel. Nous notons par ailleurs que ces évaluations ont été menées entre l'image originale et l'image modifiée, et non pas seulement sur la région sélectionnée.

3.1 Floutage

Nous constatons pour un floutage que le PSNR et le SSIM obtenus se dégradent assez rapidement plus la taille du noyau augmente. L'entropie en revanche ne montre pas de très grandes différences, quoiqu'elle diminue à mesure que les différentes plages de couleur s'uniformisent.

Nous observons en revanche une assez grande cohérence structurelle entre l'image originale et résultante, ce qui reste similaire à la façon dont nous perçevons nous-mêmes la différence structurelle entre ces images.

Taille du noyau	PSNR (dB)	SSIM	Entropie (bits/canal/pixel)
3	29,41	0,988819	7,25342
15	20,5032	0,908661	7,31645
31	17,485	0,817864	7,18596

TABLE 1 – Évaluation d'un floutage

3.2 Masquage

Le masquage est l'opération d'obscuration de notre image qui possède la plus grande capacité d'anonymisation, puisque nous pouvons que difficilement reconnaître l'objet masqué, et qu'il n'est également pas possible de revenir en arrière. Cette observation fait sens en ce qui concerne le PSNR et la fidélité structurelle entre l'image originale et sa version masquée, puisque nous avons à présent aucune dépendance entre les pixels de ces deux versions. Du fait de la nature même du masquage qui vise à remplir une région de manière uniforme, l'entropie obtenue est donc très faible.

Nous voyons par ailleurs qu'avec un masque dont la couleur est fortement présente dans l'image, nous obtenons un meilleur PSNR et SSIM, alors qu'avec des valeurs de couleur extrêmes telles que noir ou blanc, ces métriques sont minimales.

Couleur du masque	**PSNR (dB)**	**SSIM**	**Entropie (bits/canal/pixel)**
(0, 0, 0)	7.39917	0.29955	3.1291
(255, 255, 255)	7.0538	0.385071	3.39985
(212, 129, 118)	12.979	0.592131	3.51944

Table 2 – Évaluation d'un masquage

3.3 Pixélisation

Tout comme un floutage, la pixélisation vise à réduire l'information présente dans la région que l'on souhaite masquer à partir d'un paramètre, la taille des pixels, qui contrôle l'intensité de l'anonymisation. Il est évident que des tailles de pixels très faibles correspondent à un léger sous-échantillonnage de notre région, mais des valeurs plus élevées rendent l'objet de notre pixélisation plus difficile à interpréter, quoique nous arrivons tout de même à en tirer l'information sur les couleurs présentes dans l'image. L'entropie se réduit assez rapidement puisque nous réduisons tout simplement la palette de couleurs.

Taille des pixels	**PSNR (dB)**	**SSIM**	**Entropie (bits/canal/pixel)**
5	23,0618	0,954346	7,21062
10	20,0991	0,903503	7,0881
20	17,1824	0,822722	6,62551

Table 3 – Évaluation d'une pixélisation

3.4 Chiffrement

Tout comme un masquage, le chiffrement ne possède aucune dépendance sémantique entre les données originales et les données chiffrées par AES, ce qui lui confère un PSNR et un SSIM médiocre. À l'inverse du masquage en revanche, l'entropie présentée dans le tableau 4 reste très haute, ce qui est typique d'un chiffrement comme AES.

Clé	**PSNR (dB)**	**SSIM**	**Entropie (bits/canal/pixel)**
`abcdef0123456789`	10,3756	0,395571	7,52861
`0123456789abcdef`	10,3607	0,393459	7,52709
`EncryptionKey128`	10,4285	0,401297	7,52728

Table 4 – Évaluation d'un chiffrement