

Optimiser ses images avec le «Dithering»

Afin de diminuer au maximum le poids de ces images, le site de [LOW?TECH MAGAZINE] à mit en place une ancienne technique de compression d'image appelée « diffusion d'erreur ».

Cette technique repose sur deux grand principes :

La «quantification des couleurs»

Une image habituel est composé de pixel pouvant avoir nimporte quel couleur codé sur trois canneaux : **Rouge, Vert, Bleu**, avec pour chaque une **valeur allant de 0 à 255**.

L'**indexation d'image** est une technique qui consiste à réduire les couleurs de l'image à une **palette réduite**.

Cette technique, aussi nommée «**Color Quantization**» économise beaucoup de poids, **mais laisse place à moins de nuances**.

L'équipe de [LOW?TECH MAGAZINE] a fait le choix d'utilier 4 nuances de gris, et d'y appliquer un filtre CSS pour remplacer le blanc par une couleur : `mix-blend-mode: hard-light;`

Par contre, si on se contente d'indexer les couleurs d'une image dans Gimp ou Photoshop, le rendu visuel ne sera pas forcément optimal, surtout en monochrome.

Le «tramage»

Pour améliorer la visibilité de l'image avec le moins de couleurs possibles, une bonne astuce est d'utiliser des trames «dithering».

Vu a une certaines distance, la quantité de noir et de blanc se mélangent et sont percu comme un gris.

Il existe plusieurs algorithme de dithering, [imagemagick](#) permet entre autre d'en appliquer en ligne de commande.

R&D de compression

J'ai voulu expérimenter différentes options d'[imagemagick](#) pour comparer les résultat tant visuellement que sur le poids du fichier final.

- Je suis partis de cette [image source](#).
- Le poid des images s'affichent au survol.

Quantification des couleurs

```
# Resize & 2 grey : 19KB
convert gare.jpg -resize 400x600 -colors 2 -colorspace gray -normalize gare1.gif

# Resize & 3 grey : 24KB
convert gare.jpg -resize 400x600 -colors 3 -colorspace gray -normalize gare2.gif

# Resize & 3 colors : 25KB
convert gare.jpg -resize 400x600 -colors 3 -normalize gare3.gif

# Resize & 4 colors : 30KB
convert gare.jpg -resize 400x600 -colors 4 -normalize gare4.gif
```









Trammage en noir et blanc

```
# Resize & Ordered Dither 2 : 8KB
```

```
convert gare.jpg -resize 400x600 -colorspace gray -ordered-dither o2x2 gare5.gif
```

```
# Resize & Ordered Dither 3 : 10KB
```

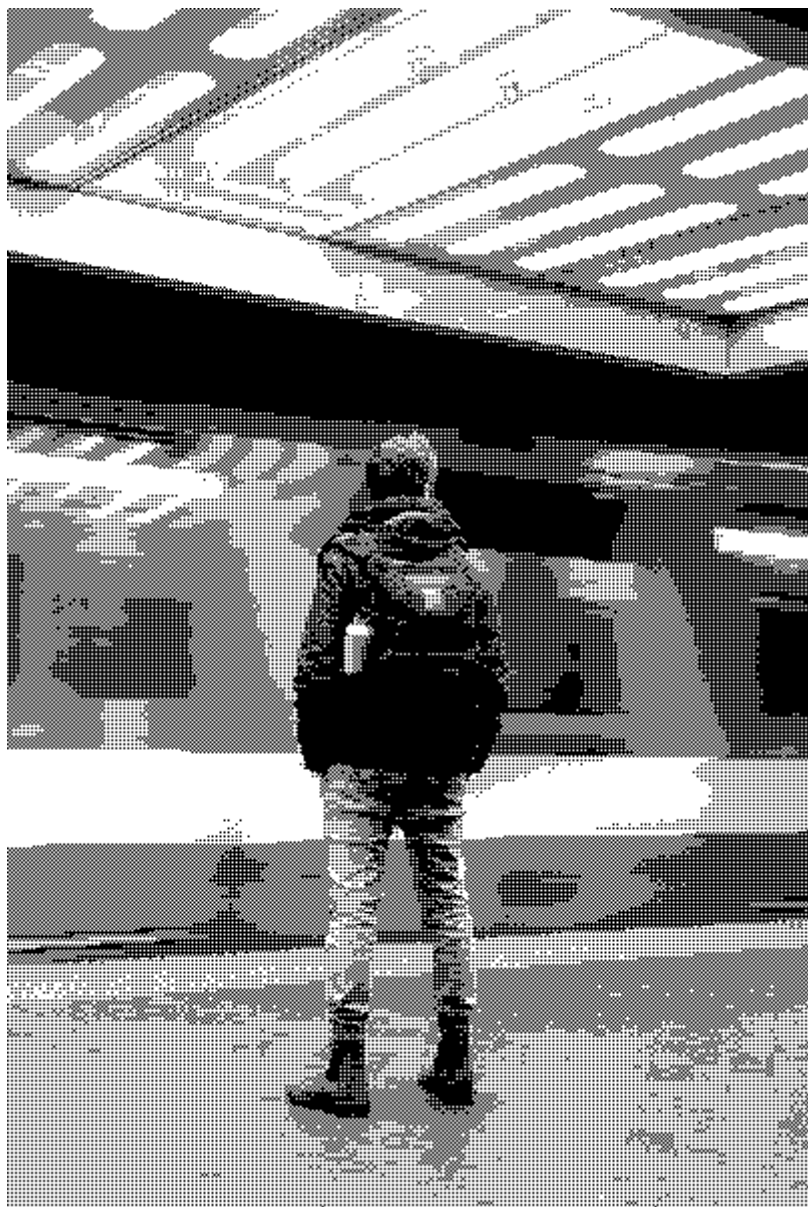
```
convert gare.jpg -resize 400x600 -colorspace gray -ordered-dither o3x3 gare6.gif
```

```
# Resize & Ordered Dither 4 : 11KB
```

```
convert gare.jpg -resize 400x600 -colorspace gray -ordered-dither o4x4 gare7.gif
```

```
# Resize & Ordered Dither 8 : 12KB
```

```
convert gare.jpg -resize 400x600 -colorspace gray -ordered-dither o8x8 gare8.gif
```









Trammage en couleurs

```
# Resize & Ordered Dither 2 : 20KB
convert gare.jpg -resize 400x600 -ordered-dither o2x2 gare9.gif

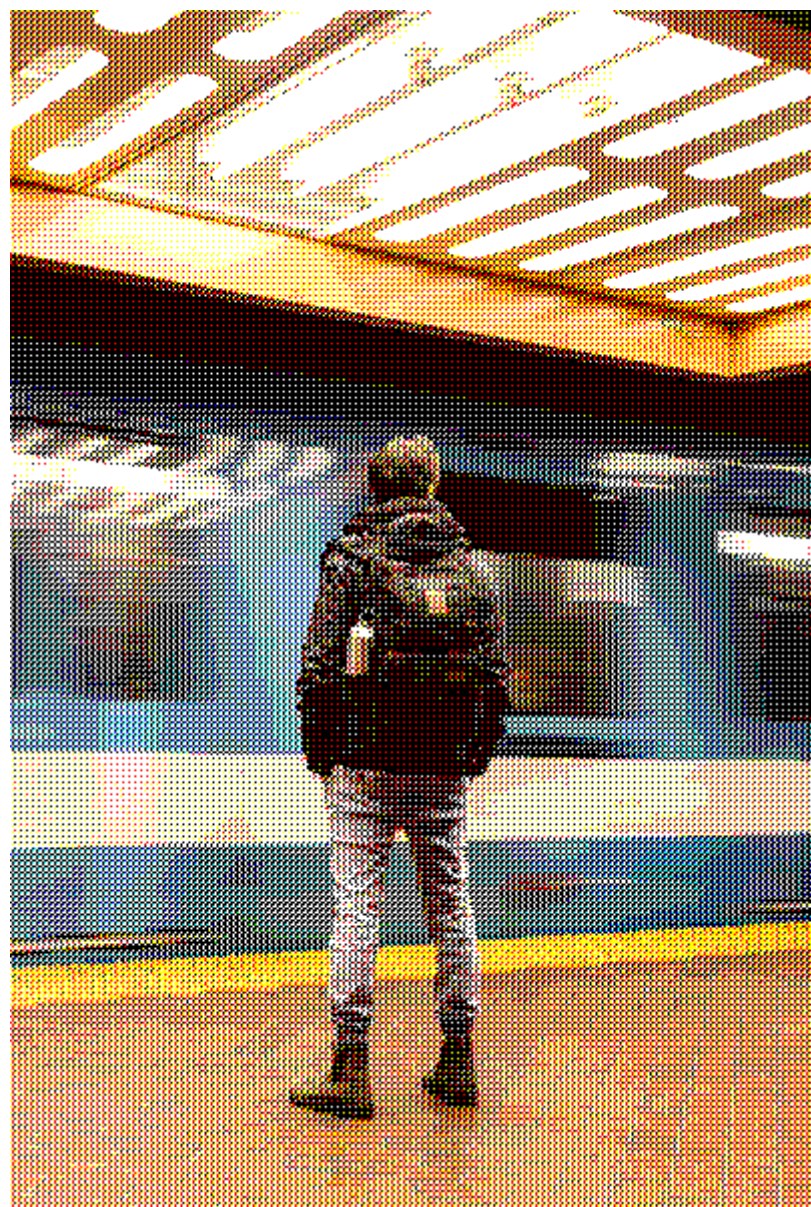
# Resize & Ordered Dither 3 : 23KB
convert gare.jpg -resize 400x600 -ordered-dither o3x3 gare10.gif

# Resize & Ordered Dither 4 : 25KB
convert gare.jpg -resize 400x600 -ordered-dither o4x4 gare11.gif

# Resize & Ordered Dither 8 : 27KB
```

```
convert gare.jpg -resize 400x600 -ordered-dither o8x8 gare12.gif
```









Comparaison

Les [trammages en noir et blanc](#) donnent les meilleur compression :

- Ordered Dither 2 : 8KB Pas assez de détails
- Ordered Dither 3 : 10KB Beaucoup mieux
- Ordered Dither 4 : 11KB Très bon comprimé
- Ordered Dither 8 : 12KB Les trames commencent à être trop visible

Suivi par :

- La quantification en 2 couleurs : 19KB pas du tout détaillée.
- Le tramage coloré Ordered Dither 2 : 20KB même niveau de détail.

Les autres vont de 23KB à 30KB ce qui commence à faire beaucoup comparé au 10KB totalement correcte que l'on a eu précédemment.

Le meilleur compromis taille et esthétique se place donc entre :

- Ordered Dither 3 10KB qui fait des trames en forme de taches
- Ordered Dither 4 11KB qui fait des trames en forme de croix

Je vais décliner ces réglages avec différentes images sources pour avoir un avis plus avisé.

Sources : [gare.jpg](#) ; [matcha.jpg](#) ; [bike.jpg](#) ; [desktop.jpg](#) ;

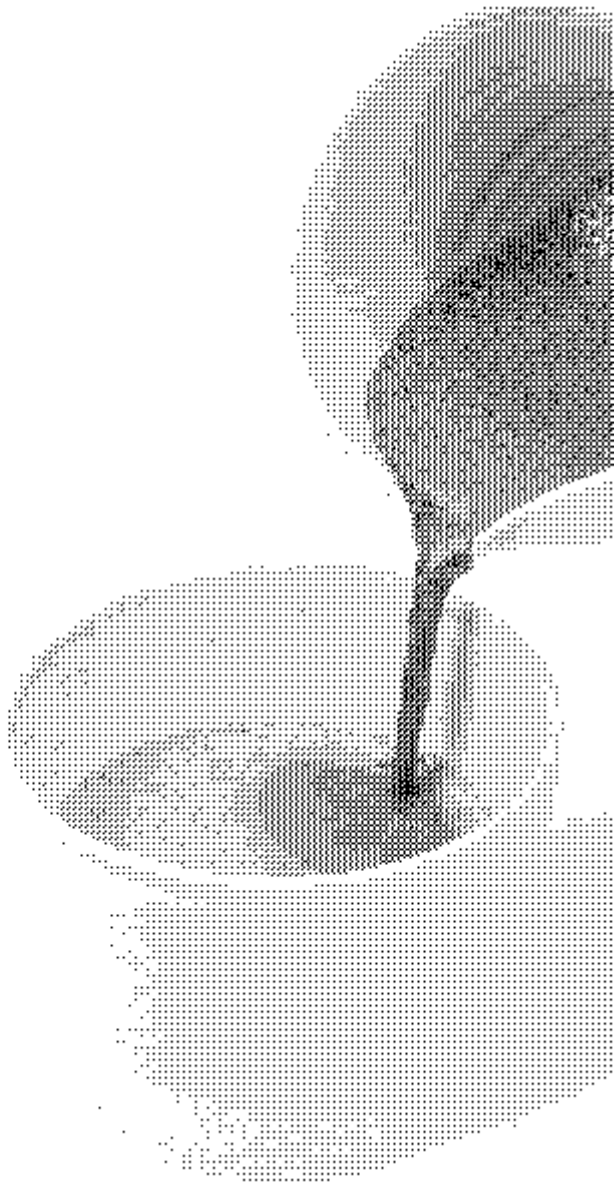
Déclinaison





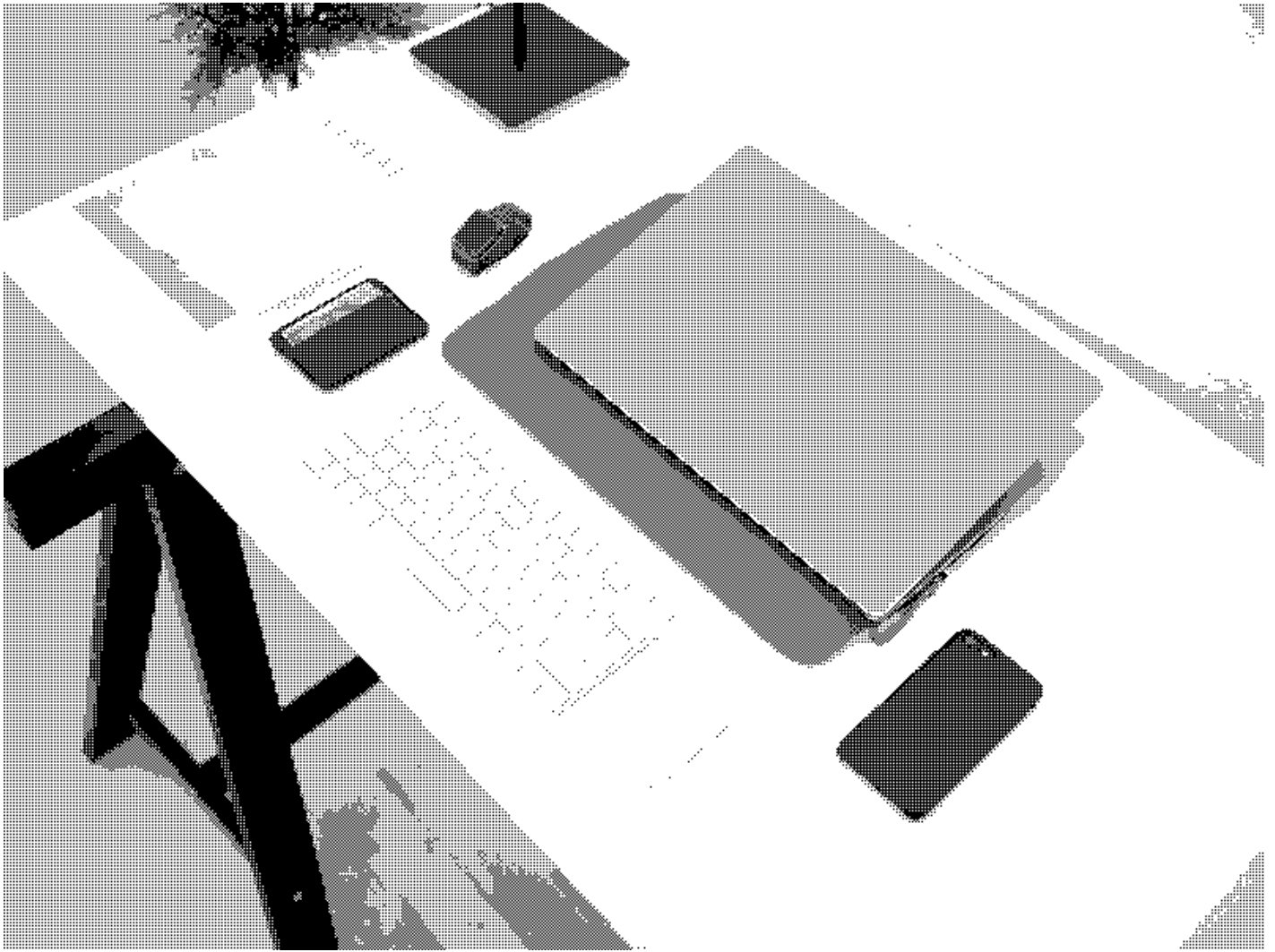
10KB & 11KB





4KB & 4KB







10KB & 11KB

Conclusion

On constate que dans certains cas, le «Ordered Dither 3» manque de détail. **Je préconise donc «Ordered Dither 4» :**

```
convert input.jpg -resize 400x600 -colorspace gray -ordered-dither o4x4 output.gif
```

Revision #9

Created 6 May 2020 19:59:55 by AntoineD

Updated 6 May 2020 22:06:01 by AntoineD