

# Machine Learning

## Practical work 03 – Unsupervised Learning

### Table of Contents

Feature extraction methods.....	2
extract_histogram.....	2
extract_color_histogram.....	2
extract_hue_histogram.....	2
Hypothèses.....	2
Expériences.....	3
1. Plages et fleurs.....	3
Comparaison des méthodes d'extraction.....	3
2. Eléphants et dinosaures.....	13
Niveaux de gris.....	14
Hue.....	15
Couleur.....	16
U-Matrix.....	16
3. Eléphants et chevaux.....	18
4. Plages, fleurs, éléphants et dinosaures.....	20

## Feature extraction methods

Toutes ces méthodes prennent en paramètre le nombre de classes à utiliser dans l'histogramme (bins en anglais). Elles travaillent sur le tableau d'image chargées au préalable. Elles retournent des tableaux d'histogrammes issus de chacune des images traitées.

### extract\_histogram

D'après le code source du fichier `WangImageUtilities.py`, cette méthode d'extraction extrait les histogrammes en nuance de gris de chaque image. Elle utilise pour ce faire la méthode `rgb2grey` du module `color` de la librairie `skimage`, qui calcule la luminance d'une image.

### extract\_color\_histogram

D'après le code source du fichier `WangImageUtilities.py`, cette méthode d'extraction extrait les histogrammes de chaque couche de couleur de chaque image. On a donc cette fois un histogramme pour chaque couche et pas simplement un histogramme de nuance de gris. Ces histogrammes sont ensuite appendus les uns aux autres.

### extract\_hue\_histogram

D'après le code source du fichier `WangImageUtilities.py`, cette méthode d'extraction extrait l'histogramme HUE (en prenant la première dimension de la méthode `rgb2hsv`). Cette méthode se concentre donc sur la teinte des images.

## Hypothèses

On peut donc imaginer que la méthode `extract_histogram` permettra une meilleure différenciation entre les images sombres et lumineuses (plus généralement qui varient en luminosité). La méthode `extract_hue_histogram` devrait permettre une meilleure différenciation des images qui varient de par la teinte. La méthode `extract_color_histogram` devrait permettre une meilleure différenciation des couleurs et de la luminosité.

# Expériences

## 1. Plages et fleurs

Dans cette première expérience, nous testons les différents paramètres de notre système (Couleurs, Niveaux de gris, Teinte, Nombre d'itérations, etc.) afin d'avoir des paramètres par défaut pour les expériences suivantes. Nous utiliserons les classes "Fleurs" et "Plages" pour cette expérience. En effet, nous émettons l'hypothèse que les deux classes sont faciles à regrouper par leurs différences en couleurs et luminosités.

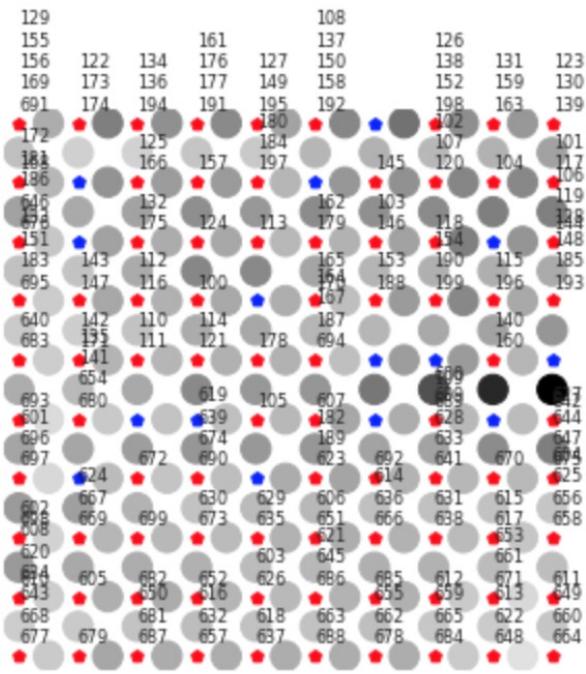
### Comparaison des méthodes d'extraction

Dans un premier temps, nous testons les "méthodes d'extraction des caractéristiques" avec une matrice de taille 10x10.

## Niveaux de gris



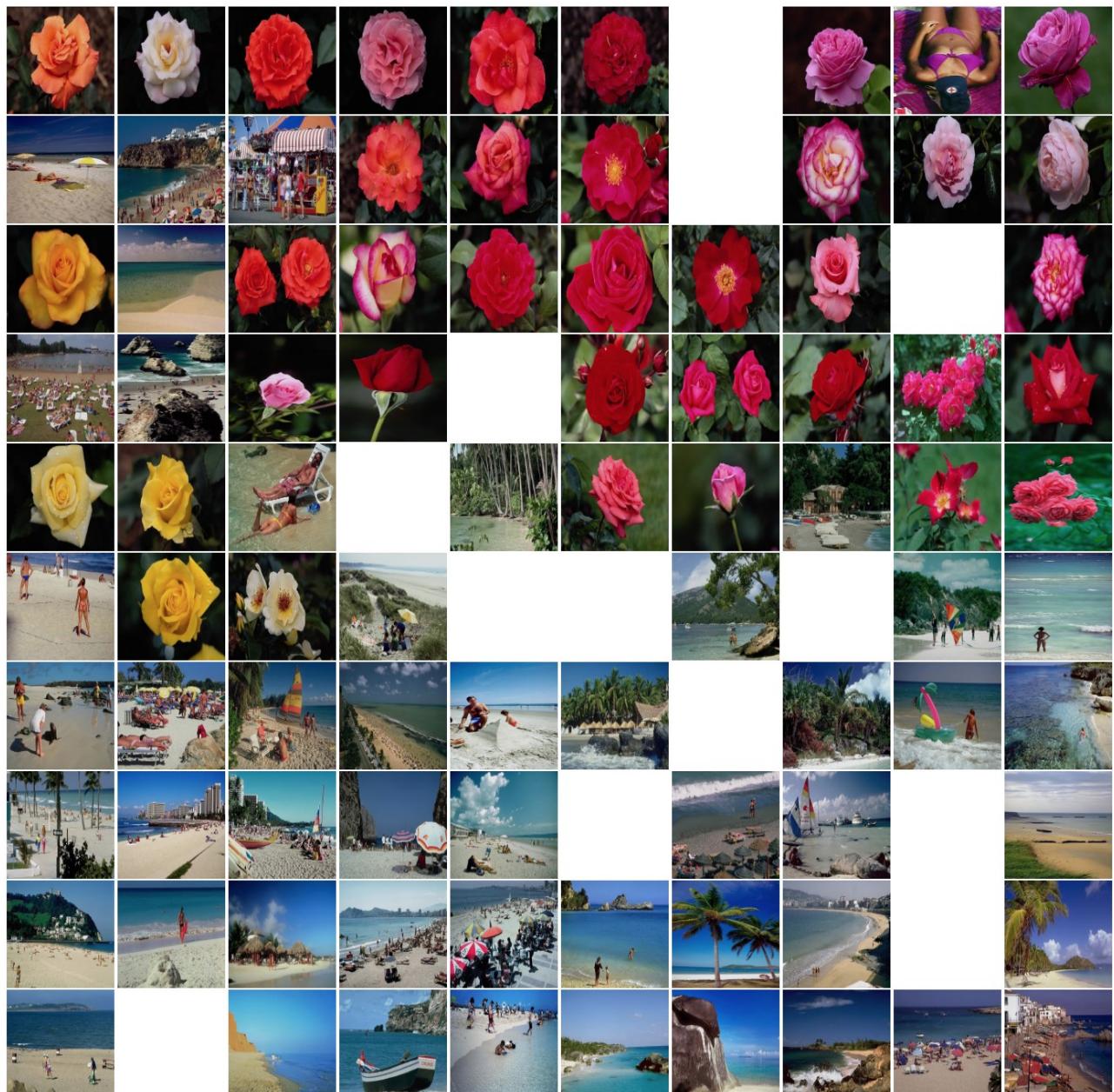
Résultat de la méthode d'extraction niveaux de gris (Plages vs Fleurs)



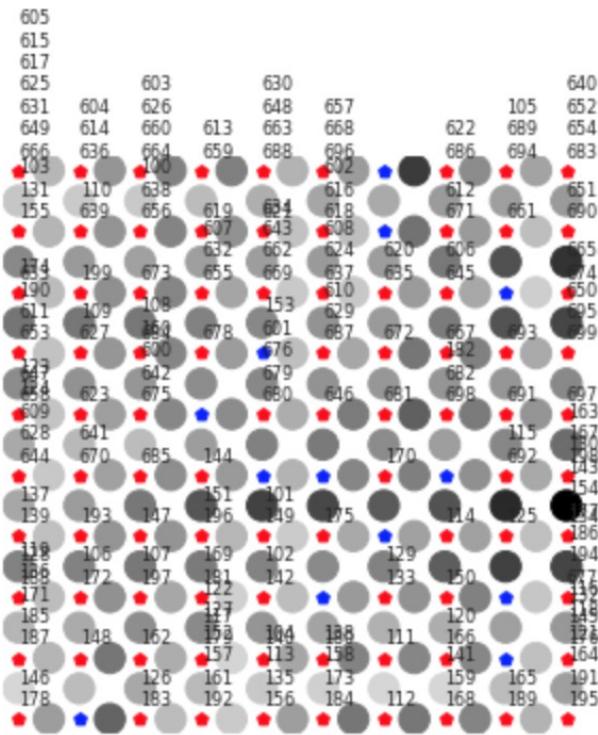
*U-Matrice de la méthode d'extraction de niveaux de gris (Plages vs Fleurs)*

Cette méthode fonctionne bien pour le regroupement des classes "Fleurs" et "Plages". En effet, les fleurs sont souvent sur des fonds très foncés alors que les plages sont très claires à cause du soleil et du sable.

## Hue

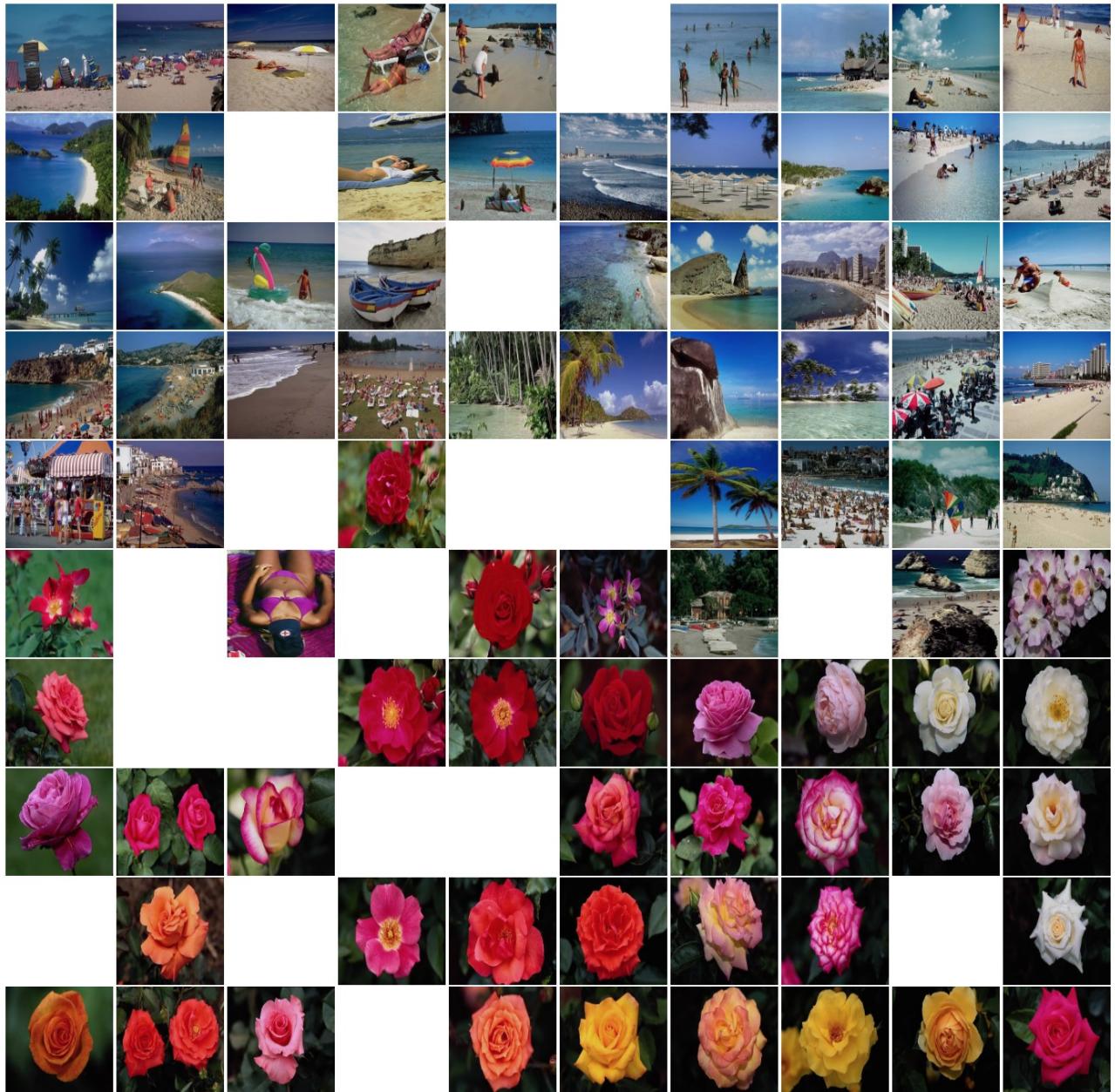


Résultat de la méthode d'extraction Hue (Plages vs Fleurs)

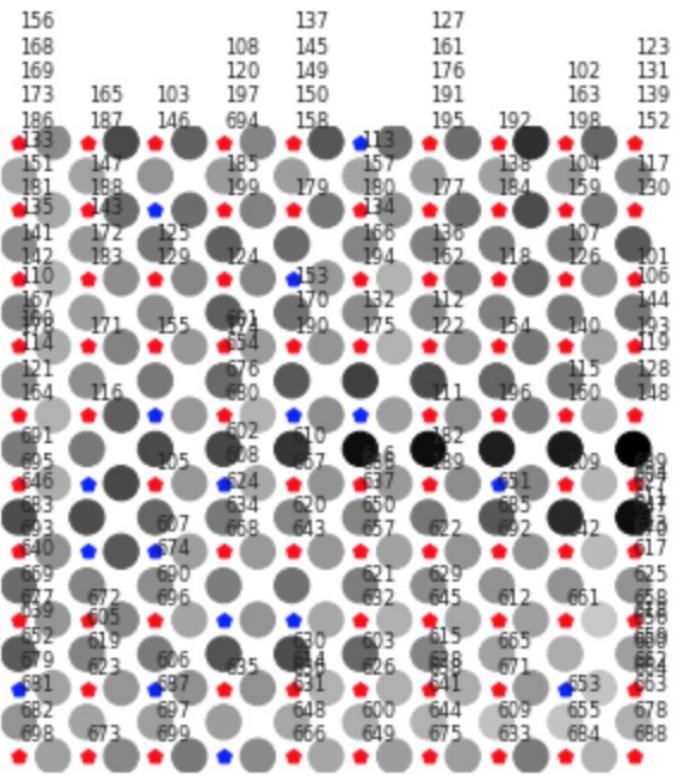


Cette méthode regroupe assez bien, sauf les fleurs de couleur jaune. En effet, cette méthode ne prend en compte que les couleurs et pas la luminosité et la saturation. Et donc les fleurs jaunes se retrouvent près des plages de teinte jaune.

## Couleur



Résultat de la méthode d'extraction de couleur (Plages vs Fleurs)



*U-Matrix de la méthode d'extraction de couleur (Plages vs Fleurs)*

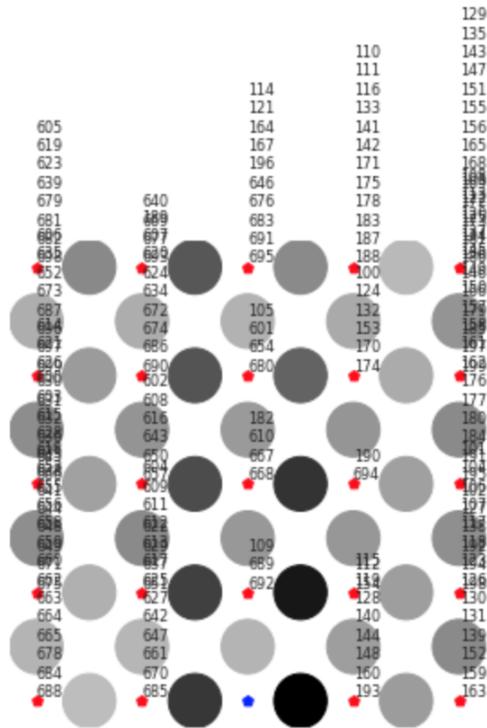
La couleur est la meilleure méthode parmi les précédentes pour regrouper les classes "Fleurs" et "Plages". Les fleurs sont très colorées alors que les plages ont souvent la même teinte de couleur (bleu ciel ou couleur sable). Nous pouvons également voir que contrairement au niveau de gris, les fleurs sont également regroupées par couleur. Et donc nous avons par exemple un cluster de fleur rouge et un cluster de fleur jaune distinguables.

## Changement de la taille de la matrice

Nous testons ensuite l'impact de la taille de la matrice sur le meilleur regroupement que nous avons obtenu ; à savoir l'extraction des histogrammes de couleur.

### Réduction

Nous présentons ici la U-Matrix résultant sur une matrice de taille 5x5 :



*U-Matrix du regroupement des couleurs sur une matrice 5x5 (Plages vs Fleurs)*

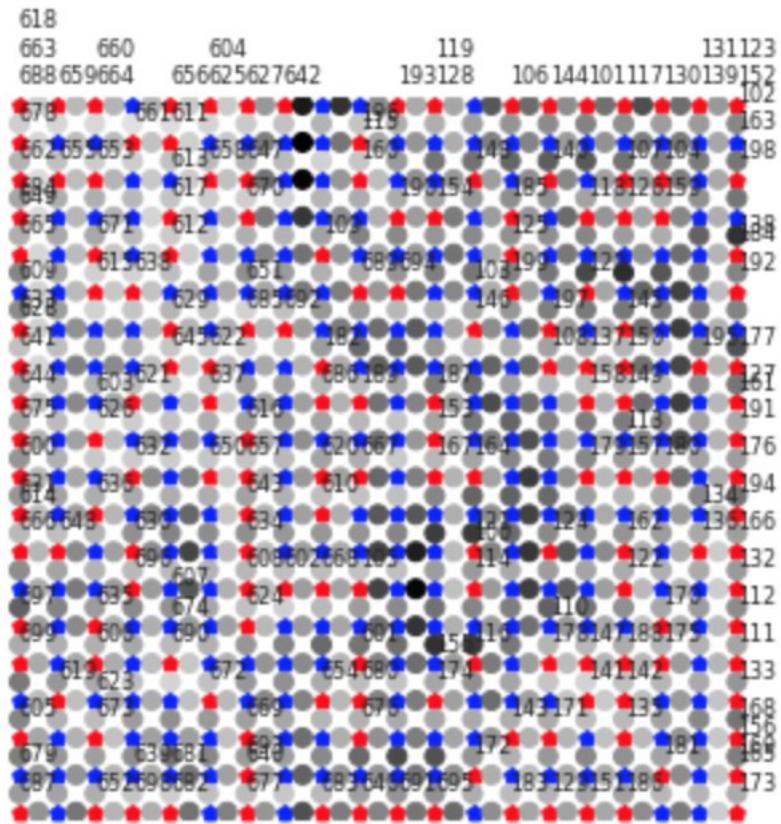
Réduire la taille de la matrice conduit à un mauvais regroupement. Nous pouvons voir que les cas extrêmes sont bien départagés mais que les cas ambigus se retrouvent tous dans le même panier. En effet deux lignes de point sombres verticales montre une bonne séparation entre les éléments de gauche et les éléments de droite. Entre ces deux lignes par contre, on retrouve des images provenant des deux classes (se référer aux numéros d'images 1XX et 6XX). Nous ne montrons pas le résultat au format HTML, mais on pouvait en effet remarquer que les regroupement au centre contenaient des fleurs et des plages (au clique de la souris).

## Augmentation

Nous présentons ensuite l'impact d'une matrice 20x20 :



Résultat de la méthode d'extraction de couleur avec une matrice  $20 \times 20$  (Plages vs Fleurs)



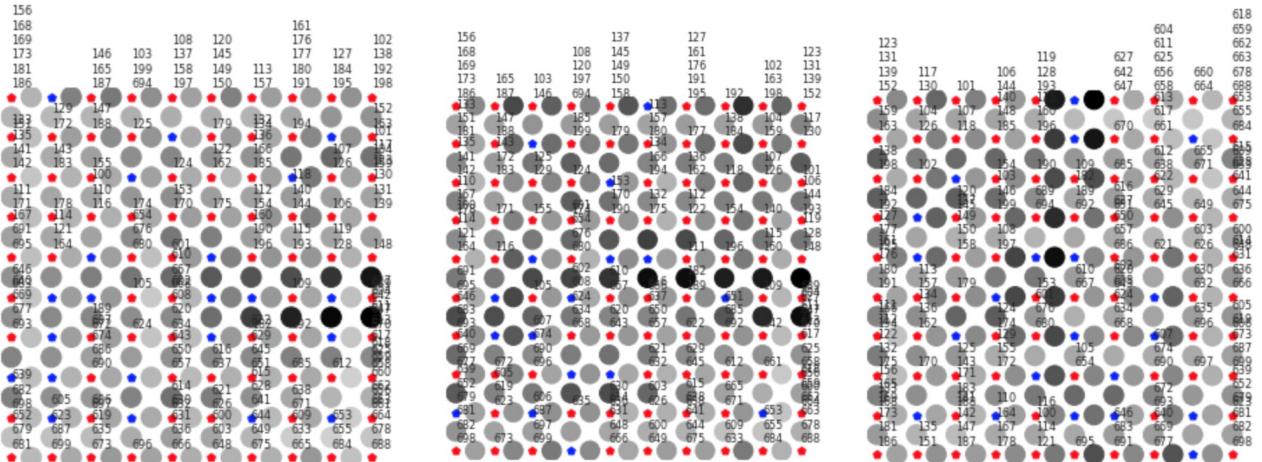
U-Matrix de la méthode d'extraction de couleur avec une matrice 20x20 (*Plages* vs *Fleurs*)

L'augmentation de la taille de la matrice permet d'avoir des séparations bien plus spécifiques. Par exemple, nous pouvons voir qu'il y a un cluster pour les fleurs roses avec un fond noir et un cluster avec des fleurs roses avec un fond vert (feuilles vertes). Mais pour cette expérience, notre but serait plutôt de regrouper des "Fleurs" et des "Plages". Ainsi, augmenter la taille de la matrice n'apporte pas de plus value mais ajoute de la complexité. Les résultats sont en effet plus difficiles à appréhender.

## Changement du nombre d'itérations

Par défaut, nous avons utilisé 100 itérations pour les expériences précédentes. Dans cette section, nous allons tester le nombre d'itération avec la méthode de couleur qui entraîne un bon regroupement pour voir si l'augmentation ou la réduction d'itération améliore le système

On compare ci-dessous trois U-Matrix (10 itérations, 100 itérations et 1000 itérations).



10 itérations

100 itérations

1000 itérations

### 10 itérations

Comparé à 100 itérations, réduire le nombre d'itération conduit à un moins bon regroupement. On remarque en effet moins de points foncés et sur les images HTML, on peut voir que les regroupement par couleur au sein des fleurs sont moins évidents.

### 1000 itérations

Augmenter le nombre d'itération n'améliore pas significativement le regroupement. On observe pas plus de points foncés et les résultats d'images HTML sont comparables à 100 itérations.

**Pour conclure cette première expérience ; pour regrouper les "Fleurs" des "Plages", nous pouvons utiliser les couleurs comme méthode d'extraction avec une matrice de taille 10x10 et 100 itérations. L'augmentation de la taille de la matrice (passé un certain seuil) ou des itérations n'améliore plus significativement les résultats.**

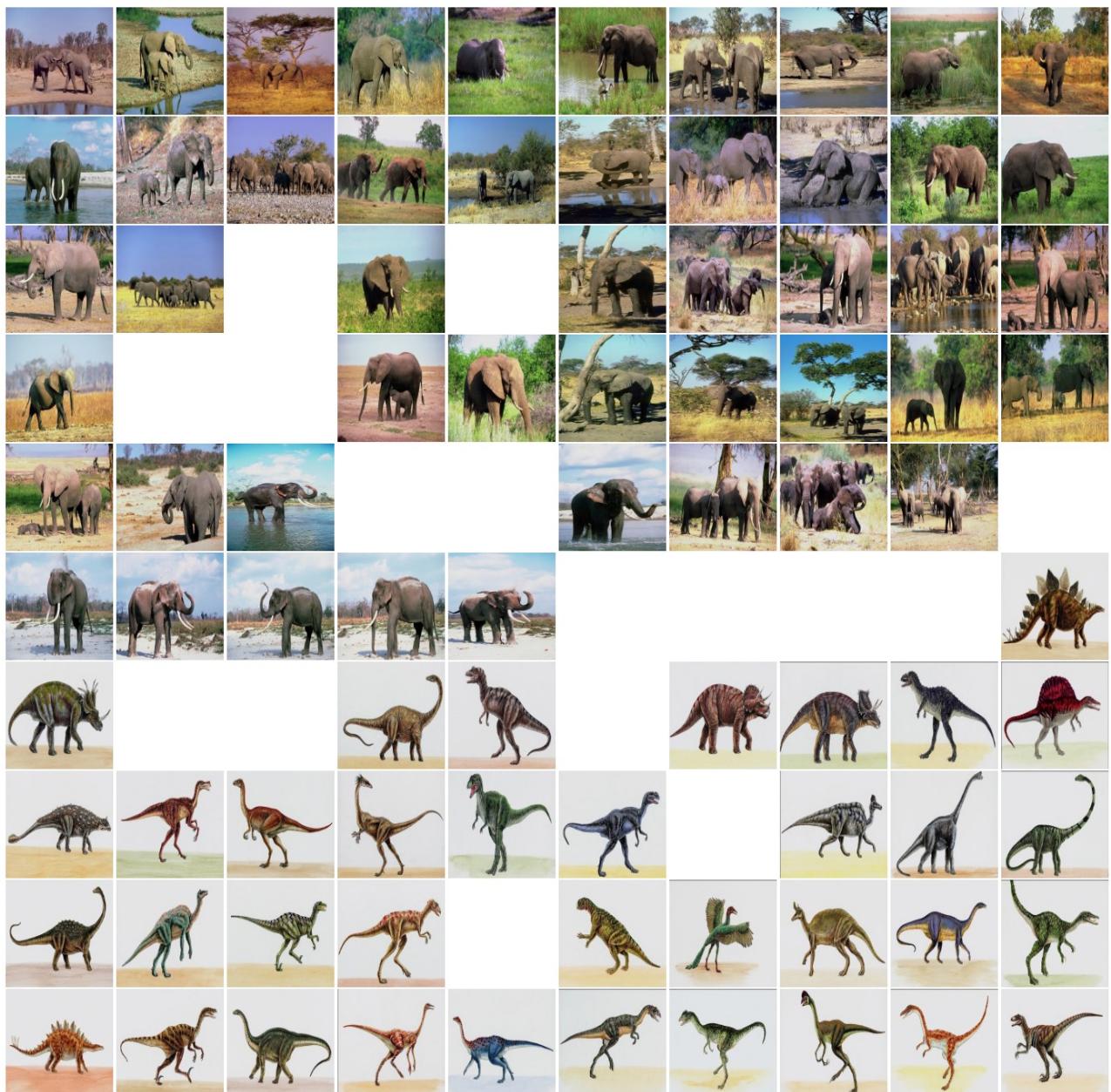
## 2. Eléphants et dinosaures

Pour cette deuxième expérience, nous testons les classes "Eléphants" et "Dinosaures". Bien que les dinosaures et les éléphants se ressemblent dans les images (grands animaux, couleur de peau, morphologie, etc.), nous faisons l'hypothèse que le système trouvera des cluster plus ou moins bien distincts parce que la classe "Dinosaure" est composée de schémas dessinés alors que la classe

"Eléphants" sont des photographies. De plus, les dinosaures sont tous sur fond blanc, ce qui simplifie la tâche.

Afin de valider ou invalider notre hypothèse, nous allons tester avec les différentes méthodes d'extraction des caractéristiques sur des matrices 10x10 avec 100 itérations:

## Niveaux de gris



*Résultat de la méthode d'extraction de niveaux de gris (Eléphant vs Dinosaur)*

Cette méthode a un bon résultat de regroupement. Les dinosaures sont tous sur un fond blanc et un sol clair alors que les éléphants se trouvent dans un environnement comme la jungle, l'eau ou le sable. Et nous pensons que ces différences permettent d'avoir des clusters bien distinct.

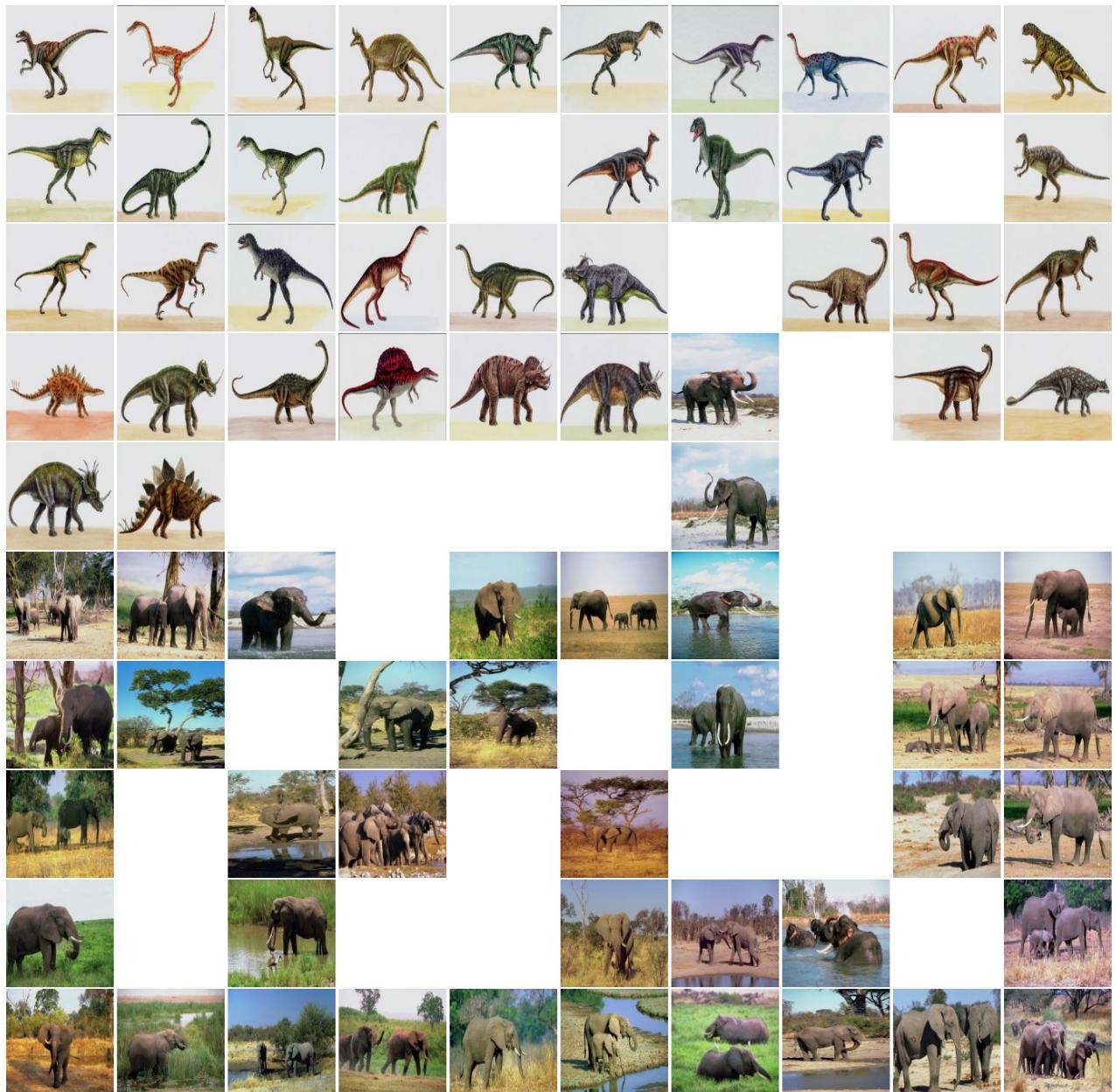
## Hue



*Résultat de la méthode d'extraction de Hue (Eléphant vs Dinosaur)*

Ici, nous pouvons voir que le regroupement est mauvais. Etant donné qu'uniquement la teinte est prise en compte, certaines images telles que les éléphants avec une teinte orange sont mal classées.

## Couleur

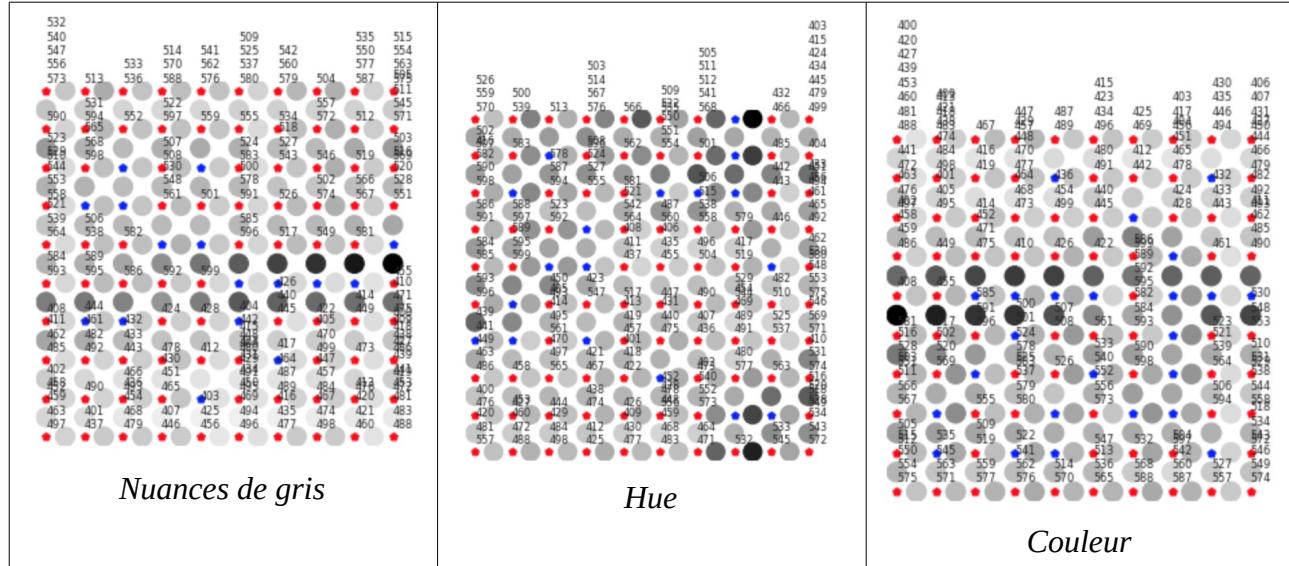


Résultat de la méthode d'extraction de couleur (Eléphant vs Dinosaur)

Comme pour la première expérience, la couleur est une très bonne méthode de regroupement. Nous pensons que ceci dû au fait que les dinosaures se ressemblent tous avec leurs couleurs unies et non-réalistes alors que les éléphants se retrouvent souvent avec des couleurs réalistes et vives. La différence se situe bien sûr également dans le fond des images.

## U-Matrix

Si l'on compare les U-Matrix, on remarque les même tendance assez facilement.



On remarque une nette séparation de points foncés pour les nuances de gris et la couleur tandis qu'il n'y a pas de tendance de séparation distincte pour l'Hue.

**Pour conclure cette expérience, l'hypothèse que nous avons établie dans un premier temps semble correcte. Bien que la teinte ne permet pas une bon regroupement, de manière générale, le système s'en sort bien.**

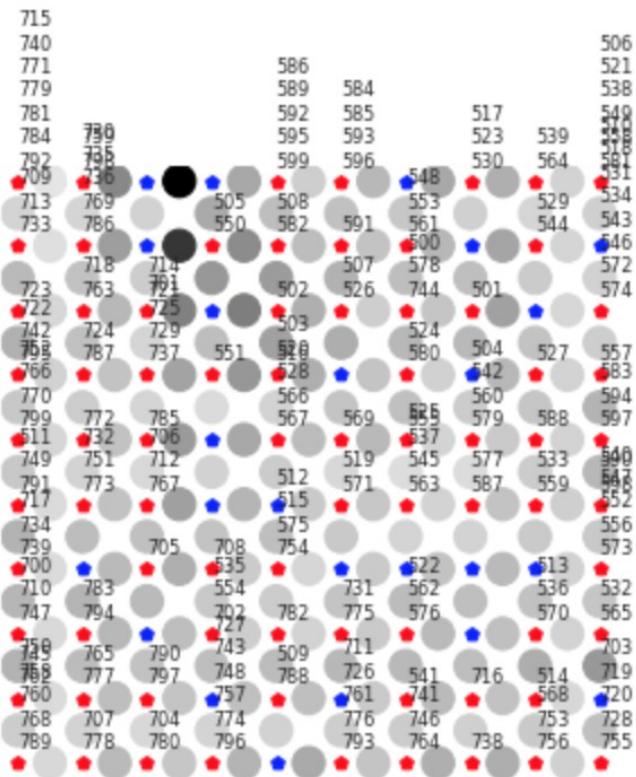
### 3. Eléphants et chevaux

En réalisant les expériences précédentes, nous nous sommes rendu compte que les couleurs de fond jouaient peut être un rôle primordial dans le regroupement des classes. Afin de tester cette hypothèse, nous avons testé la classe "Eléphant" et "Cheval" (deux animaux à l'apparence relativement semblable également). Ces deux classes sont donc intéressantes de part leurs décors plus ou moins semblables.

Nous présentons ici le résultat de la méthode d'extraction de couleur qui s'en est le mieux sortie. Cependant, quel que soit la méthode que nous utilisons, la regroupement n'est pas optimal. Les résultats des deux autres méthodes sont semblables



*Résultat de la méthode d'extraction de couleur (Eléphant vs Cheval)*



*U-Matrix de la méthode d'extraction de couleur (Eléphant vs Cheval)*

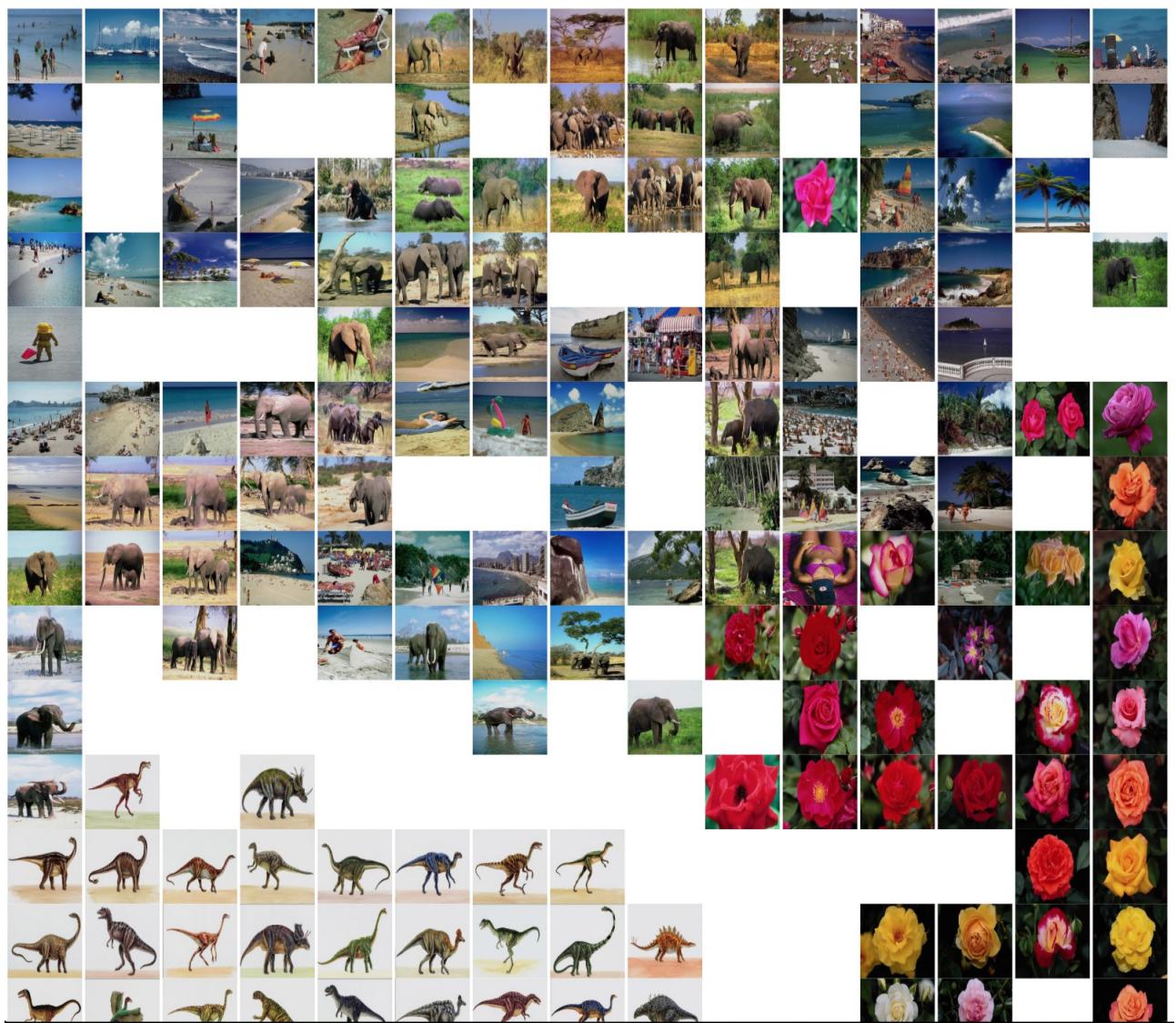
On voit bien qu'il n'y a pas de séparation claire entre différents clusters. On en conclut donc que le fond joue un rôle capital pour permettre le regroupement cohérent des classes. C'est en effet logique car sur beaucoup d'images, l'environnement du sujet occupe une bonne portion de l'image.

## 4. Plages, fleurs, éléphants et dinosaures

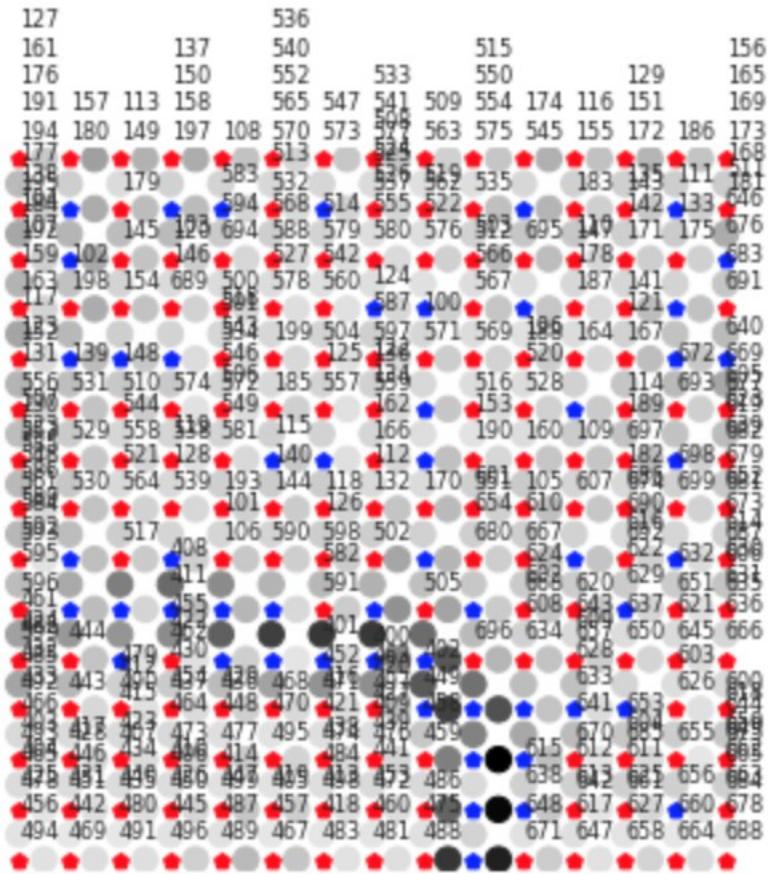
Nous avons finalement décidé de comparer quatre classes ensemble afin d'observer le comportement de l'algorithme. Nous avons décidé de prendre deux couples de classes qui se séparent très bien (plage et fleur, et éléphant et dinosaure) et d'observer le comportement entre les couples que nous n'avons pas testé.

Nous émettons l'hypothèse que les séparations distinctes observées précédemment devraient être retrouvées. Nous pensons néanmoins que si certaines des classes se ressemblent trop (les éléphants et les plages par exemple), cela pourrait rendre la tâche plus difficile à l'algorithme.

Nous présentons ici le résultat pour la méthode d'extraction de couleur.



*Résultat de la méthode d'extraction de couleur (Plage, Dinosaur, Fleur, Eléphant)*



*U-Matrix de la méthode d'extraction de couleur (Plage, Dinosaur, Fleur, Eléphant)*

Comme on peut le voir sur les deux images ci-dessus, on observe une nette distinction entre les dinosaures et les fleurs, entre les fleurs et les plages et entre les dinosaures et les éléphants.

Cependant les plages et les éléphants se mélangent de manière assez significative. Nous confirmons nos attentes concernant cette expérience.