

FICHE PROFESSEUR● **Niveau et durée :**

Seconde en SNT, séance d'une heure.

● **Objectif pédagogique :**

Manipuler la notion de composantes RGB, se repérer sur une image, utiliser des affectations, des instructions conditionnelles et des boucles « for » pour traiter une image.

● **La situation-problème :**

Dans cette activité, on dispose d'une banque de photos de différents canards. On souhaite apprendre à l'ordinateur à reconnaître un canard sur une photo et ensuite compter le nombre de canards sur chacune des photos.

● **Pré-requis :**

Les élèves doivent avoir vu les affectations, les instructions conditionnelles et la boucle *for* sur Python. Pour le traitement d'images, les élèves doivent avoir vu les instructions : *open*, *size*, *getpixel*, *putpixel*, *show* et doivent savoir ce que sont les composantes RGB d'un pixel.

Nous avons réalisé préalablement deux séances de découverte de traitement de l'image (environ 3 h).

Lien Capitale vers l'activité : <https://capitale2.ac-paris.fr/web/c/0457-570552>

● **Les consignes et la réalisation attendue :**

L'activité est décomposée en trois parties.

Dans la partie A, les élèves sont sur table et doivent proposer une méthode pour pouvoir détecter un canard sur une photo.

Dans la partie B, on s'intéresse à quatre couleurs présentes sur les canards et on compte le nombre de pixels de chacune de ces couleurs. Le but est de déterminer le nombre moyen de pixels de chacune de ces quatre couleurs sur une photo de canard.

Les élèves se connectent à leur ENT et ouvrent le fichier Capytale ou le fichier Edupython (sans oublier d'enregistrer les images et le fichier python dans le même dossier).

```
from PIL import Image
photo=Image.open('canard1.bmp') #Remplacer ensuite canard1 par canard2, ....., canard9
compteur1=0 #Aucun pixel de couleur (224,199,163)
compteur2=0 #Aucun pixel de couleur (188,146,103)
compteur3=0 #Aucun pixel de couleur (122,96,70)
compteur4=0 #Aucun pixel de couleur (68,44,33)
for x in range(100): #Toutes les photos auront pour dimensions 100*100
    for y in range(100):
        couleur=photo.getpixel((x,y))
        if couleur==(224,199,163):
            compteur1=compteur1+1
        if couleur==(188,146,103):
            compteur2=compteur2+1
        if couleur==(122,96,70):
            compteur3=compteur3+1
        if couleur==(68,44,33):
            compteur4=compteur4+1
print(compteur1,compteur2,compteur3,compteur4)
photo.show()
```

MIGRATION DES CANARDS

	Canard1	Canard2	Canard3	Canard4	Canard5	Canard6	Canard7	Canard8	Canard9	Moyenne
Couleur (224,199,163)	105	155	0	460	83	208	227	107	78	158
Couleur (188,146,103)	190	571	217	465	218	671	523	294	119	363
Couleur (122,96,70)	613	198	678	388	637	356	468	324	508	463
Couleur (68,44,33)	391	482	268	209	342	121	62	59	412	261

Dans la partie C, on met en place un programme qui permet de compter le nombre de canards sur une photo (plusieurs stratégies sont possibles).

```
From PIL import Image
photo=Image.open('photo1.bmp') #Remplacer photo1 par photo2, ....., photo9
nombrecanards=0 #Aucun canard ... pour l'instant
compteur1=0
compteur2=0
compteur3=0
compteur4=0
for x in range(600): #Toutes les photos auront pour dimensions 600*400
    for y in range(400):
        couleur=photo.getpixel((x,y))
        if couleur==(224,199,163):
            compteur1=compteur1+1
        if couleur==(188,146,103):
            compteur2=compteur2+1
        if couleur==(122,96,70):
            compteur3=compteur3+1
        if couleur==(68,44,33):
            compteur4=compteur4+1
#Première méthode :
nombrecanards=round((compteur1/158+compteur2/363+compteur3/463+compteur4/261)/4,0)
print("J'ai compté ",nombrecanards,"canard(s) sur cette photo.")

#Deuxième méthode :
nombrecanards=round((compteur1+compteur2+compteur3+compteur4)/(158+363+463+261),0)
print("J'ai compté ",nombrecanards,"canard(s) sur cette photo.")

#Autre méthode avec un encadrement
a=int(min(compteur1/158,compteur2/363,compteur3/463,compteur4/261))
b=int(max(compteur1/158,compteur2/363,compteur3/463,compteur4/261))
print("Il y a entre",a,"et",b,"canard(s) sur cette photo.")
```

MIGRATION DES CANARDS

● Analyse du dispositif :

Les élèves ont trouvé l'activité ludique.

La partie B n'a posé aucune difficulté aux élèves.

Dans la partie C, certains élèves ont eu des difficultés à voir comment utiliser les résultats de la partie B.

Plusieurs programmes différents ont été proposés.

● Modalités de travail :

En salle informatique, les élèves ont deux possibilités :

- soit ils utilisent les fichiers Python
- soit ils utilisent Capytale : pour cela, le professeur aura chargé les fichiers .ipynb en utilisant « charger un notebook » et aura associé les images à chaque partie, les élèves ont alors besoin de leur codes Educonnect pour se connecter à Toutatice et avoir accès aux activités.

● Dans les programmes :

Sciences numériques et technologie (SNT)	
Contenus	Capacités attendues
Affectations, variables Instructions conditionnelles Boucles bornées	Écrire et développer des programmes pour répondre à des problèmes et modéliser des phénomènes physiques, économiques et sociaux.
Traitement d'image	Traiter par programme une image pour la transformer en agissant sur les trois composantes de ses pixels.
Mathématiques	
Variables informatiques de type entier, booléen, flottant, chaîne de caractères Affectation Séquence d'instructions Instruction conditionnelle Boucle bornée (for)	Concevoir et écrire une instruction d'affectation, une séquence d'instructions, une instruction conditionnelle. Programmer, dans des cas simples, une boucle bornée Dans des cas plus complexes : lire, comprendre, modifier ou compléter un algorithme ou un programme.

● Les six compétences majeures

Compétences
- Modéliser Utiliser, comprendre, élaborer une simulation numérique ou géométrique prenant appui sur la modélisation et utilisant un logiciel.
- Représenter Choisir un cadre (numérique, algébrique, géométrique...) adapté pour traiter un problème ou pour représenter un objet mathématique.