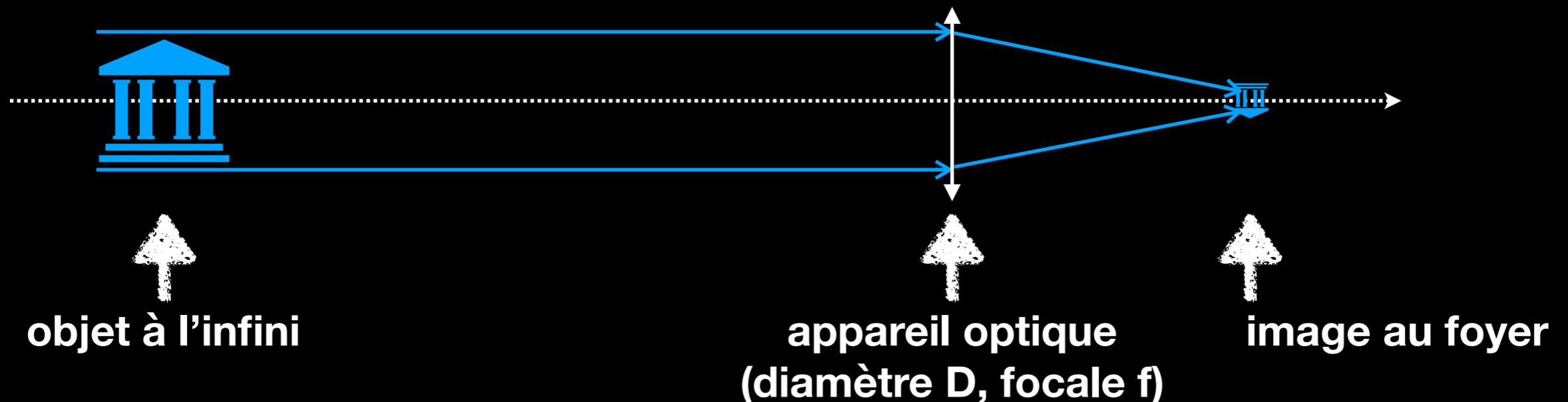


II. Analyse élémentaire d'images

6- AU FAIT, QU'EST-CE QU'UNE IMAGE ?...

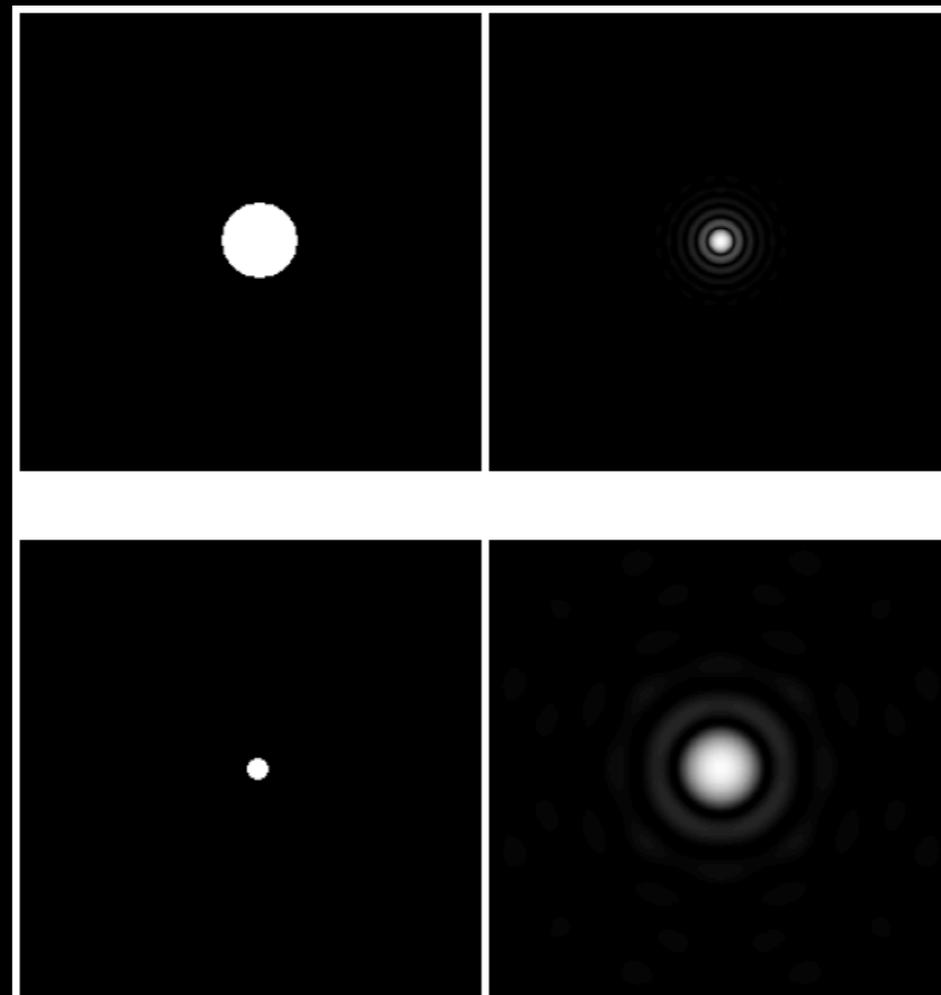
- Formation d'une image (optique) :



L'image vue (=détectée) au foyer est l'image d'un point convoluée par l'objet.

II. Analyse élémentaire d'images

Or : l'image d'un point est une « tache d'Airy », dont le cœur est de largeur à mi-hauteur $\sim \lambda f/D$ (ou $\sim \lambda/D$ en unités angulaires), avec λ la longueur d'onde.



=> plus D est grand, plus la résolution dans l'image (inversement proportionnelle à λ/D) sera importante (pour une longueur d'onde λ donnée).

II. Analyse élémentaire d'images

- Détection :

L'image précédente est ensuite détectée par un... détecteur, par exemple la matrice CCD (ou CMOS, ou EMCCD, ou autre technologie) qui équipe votre smartphone ou n'importe quel appareil « imageur » plus sophistiqué. Et cette détection est assujettie à plusieurs bruits...

- Le bruit de photons (ou *photon noise*, ou *shot noise*) qui suit une distribution de Poisson :

$p(n)$ = probabilité de détecter n photons quand N sont attendus

$$p(n) = \frac{N^n e^{-N}}{n!}, \text{ avec : } \sigma^2 = N$$

sous Matlab/Octave :

```
>> image_phot = imnoise(image_int, 'poisson');
```

II. Analyse élémentaire d'images

- Le bruit de lecture (ou *read-out noise*, *RON*) qui est un bruit ADDITIF caractérisé par une distribution de Gauss (ou 'normale') de moyenne nulle et d'écart-type σ_e :

$$p(x) = \frac{1}{\sigma_e \sqrt{2\pi}} e^{-\frac{x^2}{2\sigma_e^2}}$$

sous Matlab/Octave :

```
>> image_ron = imnoise(image_double, 'gaussian', 0.0, 0.01);
```



image en double
précision entre 0 et 1



moyenne



variance relative

II. Analyse élémentaire d'images

- Entre les deux (dans la même chaîne d'acquisition d'images), apparaissent d'autres bruits électroniques :

- pixels « chauds » et « froids » sur le CCD => bruits de type « poivre et sel »

sous Matlab/Octave :

```
>> image_snp = imnoise(image_int, 'salt & pepper', d)
```

[ATTENTION : 'salt and pepper' sous certaines versions d'Octave...]



densité de bruit
(ratio de pixels affectés)

- bruit de courant d'obscurité, bruits spécifiques à des détecteurs « exotiques » ou des applications très pointues, etc.
- « bruits périodiques » (par.ex. tramage dû à un scan ou une compression jpeg trop forte) => en fait plutôt un BIAIS qu'un véritable bruit (pas aléatoire).

II. Analyse élémentaire d'images

- **EXERCICE 5 (en mode TP...) :**
 - Prendre l'image « frog » (p.ex.). Appliquer successivement les 3 bruits principaux (bruit de photon, bruit poivre et sel, bruit de lecture), puis représenter les histogrammes consécutifs ET leur différence d'avec le précédent (utiliser PLOT).
 - Calculer et afficher minimum, maximum et moyenne de chaque image.
 - Calculer et afficher également les distances (au sens des moindres carrés) entre images et histogrammes successifs et leurs précédents.
 - Données pour les bruits : $d=5\%$, $\text{variance_RON}=1\%$.

II. Analyse élémentaire d'images

```
1 clear
2 close all
3
4 frog = imread('/Users/marcel/Documents/MATLAB/GBM/0-images/frog.jpg');
5
6 % image originale
7 frog_int = rgb2gray(frog);
8 'type de frog_int :', whos frog_int
9 min_int = min(min(frog_int)); max_int = max(max(frog_int)); mean_int=mean(mean(frog_int));
10 [n_int,x_int]=imhist(frog_int,256);
11
12 figure
13 subplot(4,3,1), imagesc(frog_int), colormap(gray), colorbar, axis('image')
14 title(['image originale'; ['min=',num2str(min_int), ' max=',num2str(max_int), ' mean=',num2str(mean_int)]]))
15 subplot(4,3,2), imhist(frog_int,256)
16 %subplot(4,3,3), plot(x_int, n_int-n_int), xlim([0 255])
17
18 % bruit de photon
19 frog_pho = imnoise(frog_int, 'poisson');
20 'type de frog_pho :', whos frog_pho
21 min_pho = min(min(frog_pho)); max_pho = max(max(frog_pho)); mean_pho=mean(mean(frog_pho));
22 dist_im = sqrt(sum(sum((frog_pho-frog_int).^2)));
23 [n_pho,x_pho]=imhist(frog_pho,256);
24 dist_hist = sqrt(sum((n_pho-n_int).^2));
25
26 subplot(4,3,4), imagesc(frog_pho), colormap(gray), colorbar, axis('image')
27 title(['photonisation', ' dist=',num2str(dist_im)] ; ['min=',num2str(min_pho), ' max=',num2str(max_pho), ' mean=',num2str(mean_pho)]]))
28 subplot(4,3,5), imhist(frog_pho,256)
29 subplot(4,3,6), plot(x_pho, n_pho-n_int), xlim([0 255]), title(['dist=',num2str(dist_hist)])
```

II. Analyse élémentaire d'images

```
31 % bruit poivre & sel
32 frog_snp = imnoise(frog_pho, 'salt & pepper', 0.05);
33 'type de frog_snp :', whos frog_snp
34 min_snp = min(min(frog_snp)); max_snp = max(max(frog_snp)); mean_snp=mean(mean(frog_snp));
35 dist_im = sqrt(sum(sum((frog_snp-frog_pho).^2)))
36 [n_snp,x_snp]=imhist(frog_snp,256);
37 dist_hist = sqrt(sum((n_snp-n_pho).^2))
38
39 subplot(4,3,7), imagesc(frog_snp), colormap(gray), colorbar, axis('image')
40 title(['poivre et sel', ' dist=', num2str(dist_im)] ; ['min=', num2str(min_snp), ' max=', num2str(max_snp), ' mean=', num2str(mean_snp)]})
41 subplot(4,3,8), imhist(frog_snp,256)
42 subplot(4,3,9), plot(x_snp, n_snp-n_pho), xlim([0 255]), title(['dist=', num2str(dist_hist)])
43
44 % bruit de lecture
45 frog_ree = double(frog_snp)/255.;
46 frog_ron = imnoise(frog_ree, 'gaussian', 0., .01);
47 'type de frog_ron :', whos frog_ron
48 %
49 frog_ron = uint8(frog_ron*255);
50 %
51 'type de frog_ron :', whos frog_ron
52 min_ron = min(min(frog_ron)); max_ron = max(max(frog_ron)); mean_ron=mean(mean(frog_ron));
53 %
54 dist_im = sqrt(sum(sum((frog_ron-frog_snp).^2)))
55 %dist_im = sqrt(sum(sum((frog_ron-double(frog_snp)/255).^2)))
56 %
57 [n_ron,x_ron]=imhist(frog_ron,256);
58 dist_hist = sqrt(sum((n_ron-n_snp).^2))
59
60 subplot(4,3,10), imagesc(frog_ron), colormap(gray), colorbar, axis('image')
61 title(['frog_\ron', ' dist=', num2str(dist_im)] ; ['min=', num2str(min_ron), ' max=', num2str(max_ron), ' mean=', num2str(mean_ron)]})
62 subplot(4,3,11), imhist(frog_ron,256)
63 subplot(4,3,12), plot(x_ron, n_ron-n_snp), xlim([0 255]), title(['dist=', num2str(dist_hist)])
```

II. Analyse élémentaire d'images

