

```
In [44]: import cv2
import os
import matplotlib.pyplot as plt
import pandas as pd
```

Based on <https://www.pyimagesearch.com/2015/09/07/blur-detection-with-opencv/>
(<https://www.pyimagesearch.com/2015/09/07/blur-detection-with-opencv/>)

```
In [15]: def calculate_blur(img):
return cv2.Laplacian(img, cv2.CV_64F).var()
```

```
In [31]: lst = os.listdir('data/toy_store_4_/resize5')
dct = {}
for img_name in lst:
    if not os.path.isdir(img_name):
        img = cv2.imread(os.path.join('data/toy_store_4_/resize5', img_name))
        dct[img_name] = calculate_blur(img)
df = pd.Series(dct)
```

```
In [59]: display(df.argmax(), df.values.max())
```

C:\Users\xinrui zhan\Anaconda3\lib\site-packages\ipykernel_launcher.py:1: FutureWarning:
The current behaviour of 'Series.argmax' is deprecated, use 'idxmax' instead.
The behavior of 'argmax' will be corrected to return the positional maximum in the future. For now, use 'series.values.argmax' or 'np.argmax(np.array(values))' to get the position of the maximum row.
"""Entry point for launching an IPython kernel.

'619.jpg'

5091.544473233414

```
In [60]: display(df.argmin(), df.values.min())
```

C:\Users\xinrui zhan\Anaconda3\lib\site-packages\ipykernel_launcher.py:1: FutureWarning:
The current behaviour of 'Series.argmin' is deprecated, use 'idxmin' instead.
The behavior of 'argmin' will be corrected to return the positional minimum in the future. For now, use 'series.values.argmin' or 'np.argmin(np.array(values))' to get the position of the minimum row.
"""Entry point for launching an IPython kernel.

'113.jpg'

95.07610622633699

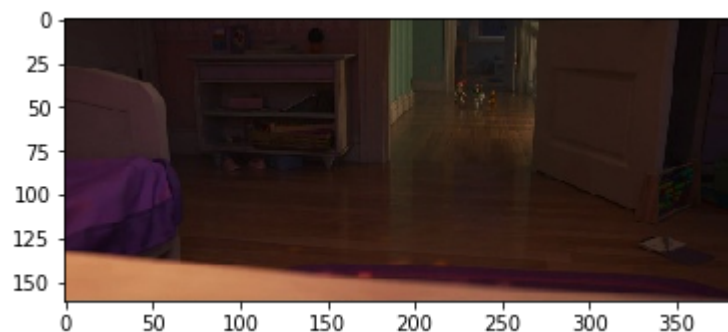
```
In [64]: plt.imshow(cv2.imread('data/toy_store_4_/resize5/619.jpg')[:, :, ::-1])
```

```
Out[64]: <matplotlib.image.AxesImage at 0x2a252457128>
```



```
In [65]: plt.imshow(cv2.imread('data/toy_store_4_/resize5/113.jpg')[:, :, ::-1])
```

```
Out[65]: <matplotlib.image.AxesImage at 0x2a2524b1cc0>
```



```
In [ ]:
```