Assignment 2 - Epipolar Geometry and 3D Reconstruction

First name: Brian

Last name: Schweigler

Matriculation number: 16-102-071

In [7]:

```
%load_ext autoreload
%autoreload 2
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt

from data import load_homogeneous_coordinates
from utils import eight_points_algorithm, right_epipole, plot_epipolar_line
```

```
The autoreload extension is already loaded. To reload it, use:
  %reload_ext autoreload
```

Part I: The Eight Point Algorithm

Load Data

In [8]:

```
data_dir = './merton_college/'
img1, img2, x1, x2 = load_homogeneous_coordinates(data_dir)
```

Estimate the fundamental matrix _F

In [9]:

```
_F = eight_points_algorithm(x1, x2)
assert np.linalg.matrix_rank(_F) == 2
print('Estimated fundamental matrix: ')
print(_F)
```

```
Estimated fundamental matrix:
[[-1.02865053e-09 -3.20313590e-07  1.31896763e-04]
 [ 9.73707906e-07  4.12485323e-08 -3.65986291e-03]
 [-4.04920255e-04  3.42996433e-03  1.43956698e-01]]
```

Plot epipoles and epipolar lines

What is an Epipolar line?

```
The plain which intersects both projection-centers (camera and object)
is called the epipolar plane.
This epipolar plane intersects both images on one line, the
epipolarline.
Only on the epipolar line, can a point on one image be connected to a
point on another image.
```

What is an Epipole?

```
The line that connects the projection centers of both camera,
intersects the image plane in the epipole.
Both epipoles may not change their position in their own image, as
long as the camera remains fixed.
The epipole is also simulatenously the depiction of the projection
centers of the other camera.
All epipolar lines of its own image pass through it.
```

In [10]:

```python
# compute the epipole
e = right_epipole(_F)
```

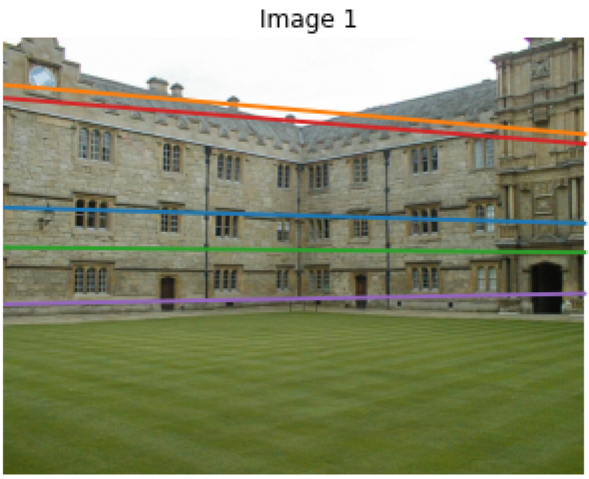
In [11]:

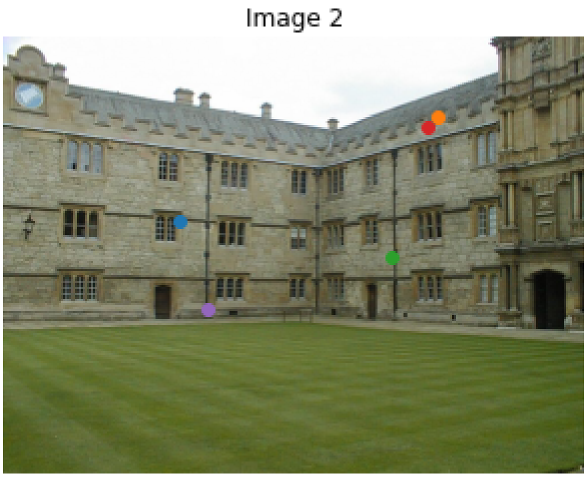
```python
plt.figure()
plt.imshow(img1)
plt.title('Image 1')

# plot each line individually, this gives nice colors
for i in range(100, 105):
    plot_epipolar_line(img1, _F.T, x2[:, i], e)

plt.axis('off')
plt.figure()
plt.imshow(img2)
plt.title('Image 2')

# plot each point individually, this gives same colors as the lines
for i in range(100, 105):
    plt.plot(x2[0, i], x2[1, i], 'o')
    plt.axis('off')
```





In [12]:

```
%matplotlib notebook
from ipywidgets import *

fig, ax = plt.subplots()
ax.imshow(img1)

fig2, ax2 = plt.subplots()
ax2.imshow(img2)

def onclick(event):
    ax.plot(event.xdata, event.ydata, 'o')
    plot_epipolar_line(img2, _F, np.array([event.xdata, event.ydata, 1]), e, ax2)
    ax2.imshow(img2)

cid = fig.canvas.mpl_connect('button_press_event', onclick)
```

In [ ]:

In [ ]: