```python
import cv2

img_location = input('Enter the image location: ')

filename = input('Enter the image name (should be in jpg extension): ')

img = cv2.imread(img_location + filename)


original_blur = cv2.GaussianBlur(img, (21, 21), 0)

gray_image = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)


inverted_gray_image = 255 - gray_image

blurred_image = cv2.GaussianBlur(inverted_gray_image, (21, 21), 0)

inverted_blurred_image = 255 - blurred_image

pencil_sketch_image = cv2.divide(gray_image, inverted_blurred_image, scale=256.0)


cv2.imshow('Original Image', img)

color_sketch = cv2.add(img, original_blur)


while True:
    i = int(input('What type of image do you want? (1=gray, 2=inverted gray, 3=blurred, 4=inverted blurred, 5=sketch, 6=color sketch, 7=all): '))


    if i == 1:
        cv2.imshow('Gray', gray_image)
        cv2.imwrite(input('Save as: '), gray_image)
    elif i == 2:
        cv2.imshow('Inverted Gray', inverted_gray_image)
        cv2.imwrite(input('Save as: '), inverted_gray_image)
    elif i == 3:
        cv2.imshow('Blurred Image', blurred_image)
        cv2.imwrite(input('Save as: '), blurred_image)
    elif i == 4:
        cv2.imshow('Inverted Blurred', inverted_blurred_image)
```

```python
            cv2.imwrite(input('Save as: '), inverted_blurred_image)
        elif i == 5:
            cv2.imshow('Final Sketch', pencil_sketch_image)
            cv2.imwrite(input('Save as: '), pencil_sketch_image)
        elif i == 6:
            cv2.imshow('Colored Sketch', color_sketch)
            cv2.imwrite(input('Save as: '), color_sketch)
        elif i == 7:
            cv2.imshow('Gray', gray_image)
            cv2.imshow('Inverted Gray', inverted_gray_image)
            cv2.imshow('Blurred Image', blurred_image)
            cv2.imshow('Inverted Blurred', inverted_blurred_image)
            cv2.imshow('Final Sketch', pencil_sketch_image)
            cv2.imshow('Colored Sketch', color_sketch)
        else:
            print('Invalid option')


        q = input('Do you want to continue? (y/n): ')
        if q.lower() != 'y':
            break


cv2.waitKey(0)
```