

**Exercise 02 for MA-INF 2201 Computer Vision WS15/16**  
**02.11.2015**  
**Submission on 08.11.2015**

1. Read the image `traffic.jpg`, build a Gaussian and Laplacian pyramid and display each layer. For the Gaussian pyramid compare two versions:
  - using `cv::pyrDown` and `cv::buildPyramid`
  - without using `cv::pyrDown` and `cv::buildPyramid`

Compute the maximal pixel-wise difference between both versions for each layer.  
*Hint:* the third parameter of `cv::pyrUp` should be used to guarantee a certain image size.

(3 Points)

2. Blend the image `apple.jpg` and `orange.jpg` using Laplacian blending<sup>1</sup> to obtain an image similar to Fig. 1.  
(5 Points)

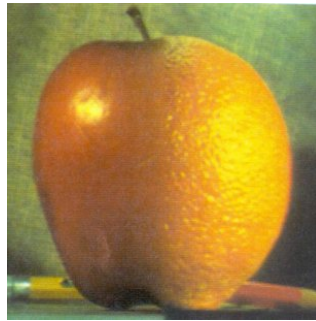


Figure 1: Blended image.

3. Read the image `traffic.jpg` and convert it into a gray image and display it.
  - Compute the gradients using `cv::sobel`.
  - Compute the gradient magnitude and gradient direction without using `cv::cartToPolar` and display both of them.

*Hint:* Use the right value `depth` for `cv::sobel` to avoid overflow.

(3 Points)

4. Read the image `traffic.jpg`, convert it into a gray image, and extract edges using `cv::Canny`. Display the gray image and the edge image.  
(2 Points)

5. Read the image `traffic.jpg`, convert it into a gray image, extract edges, compute a distance transform using `cv::distanceTransform`. Display the distance transformation. Read `sign.png`, extract the edges, and display it. Detect the two largest traffic signs in the image `traffic.jpg` by Chamfer matching and visualize the detections, as well as the accumulator (voting space). *Hint:* Rescale the template to the size of the two largest traffic signs in the image.  
(7 Points)

---

<sup>1</sup>For further details you can refer to: *P. Burt and E. Adelson. A Multiresolution Spline With Application to. Image Mosaics. ACM Trans. Graph. 1983*