178 3 Image processing

3. Compare the quality of these two algorithms.

**Ex 3.25: Feature-based morphing** Extend the warping code you wrote in Exercise 3.23 to import two different images and specify correspondences (point, line, or mesh-based) be- tween the two images.

1. Create a morph by partially warping the images towards each other and cross-dissolving (Section 3.6.3).
2. Try using your morphing algorithm to perform an image rotation and discuss whether it behaves the way you want it to.

**Ex 3.26: 2D image editor** Extend the program you wrote in Exercise 2.2 to import images and let you create a “collage” of pictures. You should implement the following steps:

1. Open up a new image (in a separate window).
2. Shift drag (rubber-band) to crop a subregion (or select whole image).
3. Paste into the current canvas.
4. Select the deformation mode (motion model): translation, rigid, similarity, afﬁne, or perspective.
5. Drag any corner of the outline to change its transformation.
6. (Optional) Change the relative ordering of the images and which image is currently being manipulated.

The user should see the composition of the various images’ pieces on top of each other.

This exercise should be built on the image transformation classes supported in the soft- ware library. Persistence of the created representation (save and load) should also be sup- ported (for each image, save its transformation).

**Ex 3.27: 3D texture-mapped viewer** Extend the viewer you created in Exercise 2.3 to in- clude texture-mapped polygon rendering. Augment each polygon with (*u, v, w*) coordinates into an image.

**Ex 3.28: Image denoising** Implement at least two of the various image denoising tech- niques described in this chapter and compare them on both synthetically noised image se- quences and real-world (low-light) sequences. Does the performance of the algorithm de- pend on the correct choice of noise level estimate? Can you draw any conclusions as to which techniques work better?

**Ex 3.29: Rainbow enhancer—challenging** Take a picture containing a rainbow, such as Figure 3.66, and enhance the strength (saturation) of the rainbow.

1. Draw an arc in the image delineating the extent of the rainbow.