

## Tutorial Augmented Reality

### Exercise 15 Putting it all together (example solution)

In the example solution, the inner loop from the tracker has been moved to the GLUT `idle` function. In order to pass tracking data to OpenGL, some of the previously local variables (like `resultMatrix`) have to be moved to global scope.

To update OpenCV windows (for debugging purposes), we still have to call `cvWaitKey` regularly. We do so at the end of the idle function. To update the OpenGL window, `glutPostRedisplay` is called directly after pose of the marker with the appropriate ID has been determined.

Note that `glLoadTransposeMatrixf` has to be used in order to convert between column-major and row-major matrices.

### Exercise 16 Background Video (example solution)

The basic steps to achieve this are:

- Set the pixel zoom and storage parameters (`glPixelStorei()`, `glPixelZoom()`).
- Store default projection matrix and disable the depth test (`glDisable()`).
- Set an orthogonal projection matrix covering the entire screen (`gluOrtho2D()`).
- Set the raster position (`glRasterPos2i()`).
- Call `glDrawPixels`, passing the pixel buffer from the `IplImage`.
- Restore the original projection matrix and re-enable the depth test.
- Load the modelview matrix and draw the snowman (as in Exercise 15).

The first step is done once during initialization, the other steps have to be repeated for each rendering cycle.

```
1 // initialisation
2 // pixel storage/packing
3 glPixelStorei( GL_PACK_ALIGNMENT, 1 );
4 glPixelStorei( GL_UNPACK_ALIGNMENT, 1 );
5 glPixelZoom( 1.0, -1.0 );
6
7 [...]
8
9 // clear buffers
10 glClear( GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT );
11
```

```
12 glMatrixMode( GL_MODELVIEW );
13 glLoadIdentity();
14
15 // draw background image
16 glDisable( GL_DEPTH_TEST );
17
18 glMatrixMode( GL_PROJECTION );
19 glPushMatrix();
20 glLoadIdentity();
21 gluOrtho2D( 0.0, width, 0.0, height );
22
23 glRasterPos2i( 0, height-1 );
24 glDrawPixels( width, height, GL_RGB, GL_UNSIGNED_BYTE, bkgnd );
25
26 glPopMatrix();
27
28 glEnable( GL_DEPTH_TEST );
29
30 // move to origin
31 glMatrixMode( GL_MODELVIEW );
32 glLoadTransposeMatrixf( resultMatrix );
```

*Note:* There are, of course, more ways to achieve this, for example textures.