

- 6.1 Point sources produce a very harsh lighting. Such images are characterized by abrupt transitions between light and dark. The ambient light in a real scene is dependent on both the lights on the scene and the reflectivity properties of the objects in the scene, something that cannot be computed correctly with OpenGL. The Phong reflection term is not physically correct.
- 6.3 If we were to take into account a light source being obscured by an object, we would have to have all polygons available so as to test for this condition. Such a global calculation is incompatible with the pipeline model that assumes we can shade each polygon independently of all other polygons as it flows through the pipeline.
- 6.7 Let  $\psi$  be the angle between the normal and the halfway vector,  $\phi$  be the angle between the viewer and the reflection angle, and  $\theta$  the angle between the normal and the light source. If all the vectors lie in the same plane, the angle between the light source and the viewer can be computed either as  $\phi + 2\theta$  or  $2(\theta + \psi)$ . Setting the two equal we get  $\phi = 2\psi$ . If the vectors are not coplanar then  $\phi < 2\psi$ .
- 6.19 Shading requires that when we transform normals and points, we maintain the angle between them or equivalently have the dot product  $p \cdot v = p' \cdot v'$  when  $p' = Mp$  and  $n' = Mn$ . If  $M^T M$  is an identity matrix, angles are preserved. Such a matrix ( $M^{-1} = M^T$ ) is called orthogonal. Rotations and translations are orthogonal but scaling and shear are not.
- 6.21 One approach is to rotate the given plane to plane  $z = 0$  and rotate the light source and objects in the same way. Now we have the same problem solved in Chapter 5, Section 10, and can rotate everything back at the end.