RSO Report #1 – BRDF Sampling

Jakub Profota

October 26, 2025

Source Code

```cpp
bool sampleDirection(const vec3 &N, const vec3 &V, vec3 &L) {
  L = vec3(0.0, 0.0, 0.0);
  double p_diff = diffuseAlbedo.average();
  double p_spec = specularAlbedo.average();
  double p_total = p_diff + p_spec;
  if (p_total <= epsilon) return false;
  p_diff /= p_total; p_spec /= p_total;
  double xi = drandom();

  if (xi < p_diff) { // Diffuse
    double xi1 = xi / p_diff; double xi2 = drandom();

    double sinTheta = sqrt(xi1); double phi = 2.0 * M_PI * xi2;
    double x_local = sinTheta * cos(phi);
    double y_local = sinTheta * sin(phi);
    double z_local = sqrt(1.0 - xi1);

    vec3 T, B;
    vec3 a = (fabs(N.x) <= epsilon) ?
      vec3(1.0, 0.0, 0.0) : vec3(0.0, 1.0, 0.0);
    B = cross(N, a).normalize(); T = cross(B, N);

    L = (T * x_local + B * y_local + N * z_local).normalize();

  } else if (xi < p_diff + p_spec) { // Specular
    double xi1 = (xi - p_diff) / p_spec; double xi2 = drandom();

    double cosAlpha = pow(xi1, 1.0 / (shininess + 1.0));
    double sinAlpha = sqrt(1.0 - cosAlpha * cosAlpha);
    double phi = 2.0 * M_PI * xi2;

    vec3 k = V;
    vec3 a = (fabs(k.x) <= epsilon) ?
      vec3(1.0, 0.0, 0.0) : vec3(0.0, 1.0, 0.0);
    vec3 i = cross(a, k).normalize();
    vec3 j = cross(k, i);

    double x_local = sinAlpha * cos(phi);
    double y_local = sinAlpha * sin(phi);
    double z_local = cosAlpha;

    vec3 R_m = (i * x_local + j * y_local + k * z_local).normalize();
    L = (N * (2.0 * dot(N, R_m)) - R_m).normalize();
  } else {
    return false;
  }
}
```
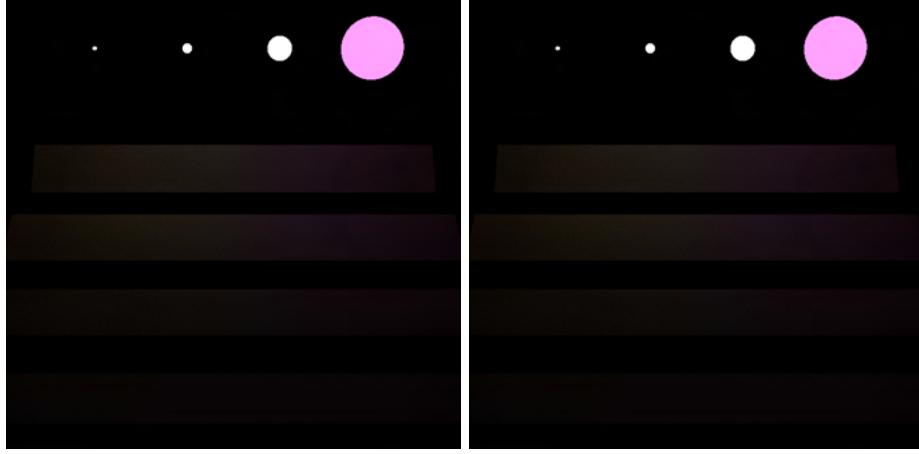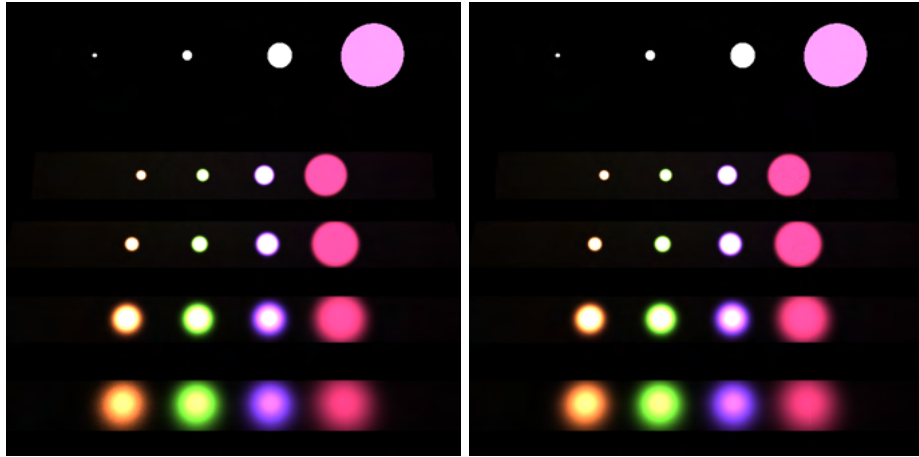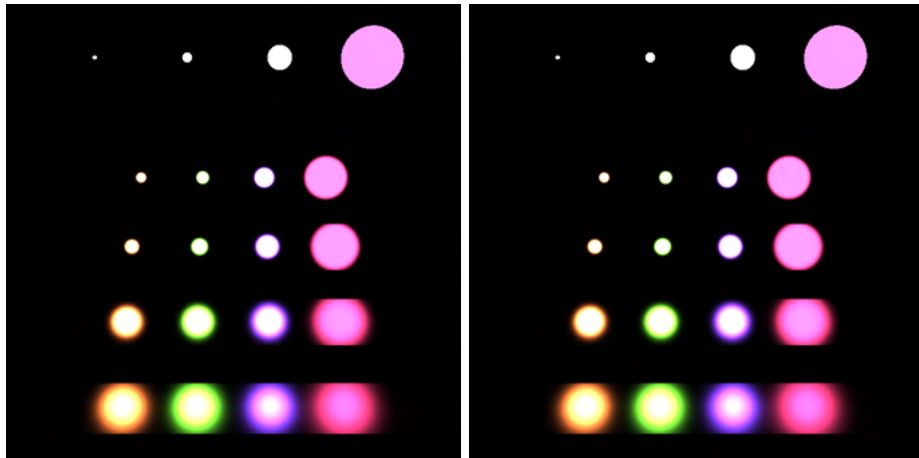
Results

The following images were rendered in 50 iterations, with 10,000 samples per iteration. The total compute time spent per image was approximately 5 hours. The left column was produced using BRDF importance sampling, while the right column represents direct illumination.

Diffuse albedo $(1, 1, 1)$, specular albedo $(0, 0, 0)$.

Diffuse albedo $(0.5, 0.5, 0.5)$, specular albedo $(0.5, 0.5, 0.5)$.

Diffuse albedo $(0, 0, 0)$, specular albedo $(1, 1, 1)$.