

# Complex Fresnel shader



## Overview

**i** With the introduction of the **metalness** parameter to the **VRayMtl** material in V-Ray Next, this texture is obsolete. The metalness implementation in the VRayMtl material is more accurate and in addition it includes proper glossy Fresnel calculations for glossy surfaces.

This shader can be used to render materials with complex index of refraction. The shader is used as a reflection color map in a **VRayMtl** material with the **VRayOSLTex** texture, with the **Fresnel** option of the V-Ray material turned off (this shader does its own Fresnel calculations). The site <http://refractiveindex.info/> has the measured n and k values for many materials like copper, gold etc.

The shader file can be found here: [complex\\_ior.zip](#)

## Parameters

**n** - the n values for the red/green/blue wavelengths (e.g. 0.65, 0.55 and 0.45 micrometers).

**k** - the k values for the red/green/blue wavelengths

## Shader code

The shader code is below.

Shader parameters

n.....	0.271	0.677	1.316
k.....	3.609	2.625	2.292

## complex\_ior.osl

```
float fresnel(float n, float k, float c) {
    float k2=k*k;
    float rs_num = n*n + k2 - 2*n*c + c*c;
    float rs_den = n*n + k2 + 2*n*c + c*c;
    float rs = rs_num/ rs_den ;

    float rp_num = (n*n + k2)*c*c - 2*n*c + 1;
    float rp_den = (n*n + k2)*c*c + 2*n*c + 1;
    float rp = rp_num/ rp_den ;

    return clamp(0.5*( rs+rp ), 0.0, 1.0);
}

shader complex_ior (
    vector n=vector(0.27105, 0.67693, 1.3164) [[ string description = "Refractive index for red, green,
blue wavelengths (e.g. for 0.65, 0.55, 0.45 micrometers)" ]],
    vector k=vector(3.6092, 2.6247, 2.2921) [[ string description = "Extinction coefficient for red, green,
blue wavelengths (e.g. for 0.65, 0.55, 0.45 micrometers)" ]],
    output color Col_Out = color(0.5)
)
{
    float thetaCos = abs(dot(-I,N));
    float red=fresnel(n[0], k[0], thetaCos);
    float green=fresnel(n[1], k[1], thetaCos);
    float blue=fresnel(n[2], k[2], thetaCos);
    Col_Out=color(red, green, blue);
}
```