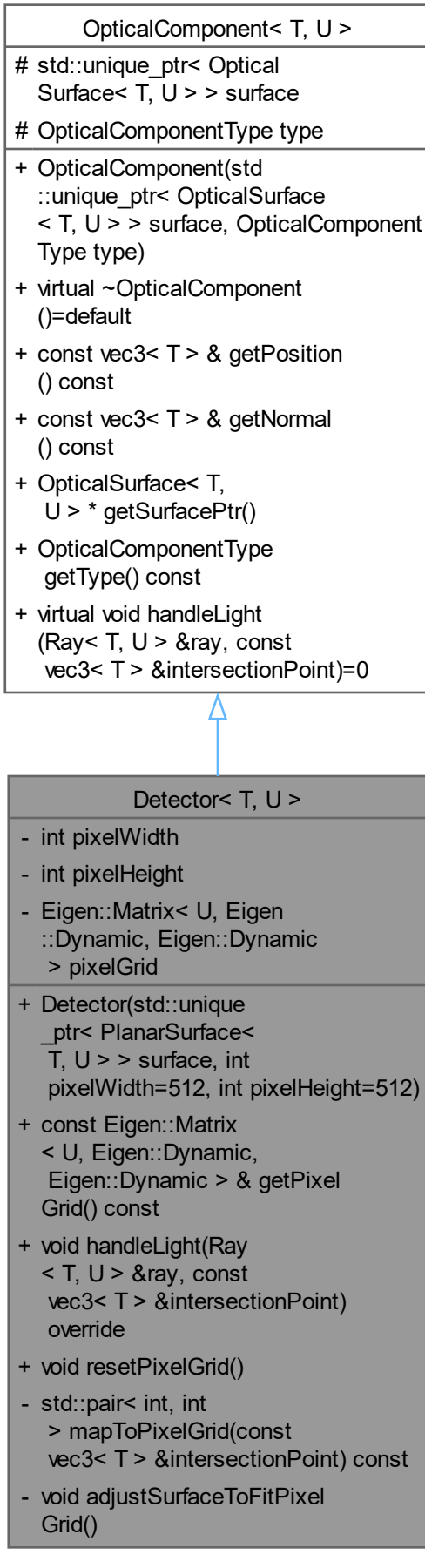
OpticalComponent< T, U >
std::unique_ptr< Optical Surface< T, U > > surface
OpticalComponentType type
+ OpticalComponent(std ::unique_ptr< OpticalSurface < T, U > > surface, OpticalComponent Type type)
+ virtual ~OpticalComponent ()=default
+ const vec3< T > & getPosition () const
+ const vec3< T > & getNormal () const
+ OpticalSurface< T, U > * getSurfacePtr()
+ OpticalComponentType getType() const
+ virtual void handleLight (Ray< T, U > &ray, const vec3< T > &intersectionPoint)=0
Detector< T, U >
- int pixelWidth
- int pixelHeight
- Eigen::Matrix< U, Eigen ::Dynamic, Eigen::Dynamic > pixelGrid
+ Detector(std::unique _ptr< PlanarSurface< T, U > > surface, int pixelWidth=512, int pixelHeight=512)
+ const Eigen::Matrix < U, Eigen::Dynamic, Eigen::Dynamic > & getPixel Grid() const
+ void handleLight(Ray < T, U > &ray, const vec3< T > &intersectionPoint) override
+ void resetPixelGrid()
- std::pair< int, int > mapToPixelGrid(const vec3< T > &intersectionPoint) const
- void adjustSurfaceToFitPixel Grid()