

# Luxoria - Dependencies Justification

Last Updated: 01/06/2025

## Justification for Dependencies

In modern application development, selecting the right dependencies is crucial to ensuring performance, reliability, and maintainability. This document outlines the rationale behind the inclusion of SkiaSharp 3.116.1, Luxoria.Algorithm.BrisqueScore (OpenCV 4.10.0 / C++23), and Sentry SDK in our WinUI 3-based application, providing justifications supported by industry standards and research.

### SkiaSharp 3.116.1

SkiaSharp is a cross-platform 2D graphics library built on Google's Skia graphics engine, widely adopted in Chrome, Android, and Flutter. It enables high-quality rendering, text shaping, and image processing, making it an essential component for modern UI development.

#### Why SkiaSharp for WinUI 3?

- **High-Performance Rendering**

SkiaSharp supports GPU acceleration and hardware-based optimizations, significantly enhancing real-time rendering performance. Google's documentation states that Skia outperforms software rasterization, making it a preferred choice for applications requiring smooth rendering ([Google Skia Docs](#)).

- **Advanced Drawing Capabilities**

SkiaSharp provides anti-aliasing, vector graphics, complex path rendering, and blending modes, surpassing WinUI 3's default rendering capabilities. These features enable the creation of visually compelling and interactive applications ([Microsoft Docs](#)).

- **Cross-Platform Consistency**

Ensures that graphics rendering remains uniform across Windows, macOS, Linux, iOS, and Android, making it future-proof for cross-platform expansion ([Mono Project](#)).

- **Seamless Integration with WinUI 3**

Supports .NET 9 and WinUI 3, making it a natural choice for modern Windows applications. It enables developers to implement rich graphical interfaces while maintaining high performance.

- **Optimized Memory Management**

Unlike GDI+ or Direct2D, SkiaSharp employs hardware-accelerated rendering techniques that reduce CPU and memory consumption, improving application efficiency and reducing power consumption ([Google Skia](#)).

- **Custom Font & Text Shaping**

SkiaSharp includes Harfbuzz for text shaping, allowing precise typography rendering, including international scripts ([Harfbuzz Project](#)).

- **Resizing Performance**

Comparaison & Benchamarks available at : <https://anthonysimmon.com/benchmarking-dotnet-libraries-for-image-resizing/>

## **Luxoria.Algorithm.BrisqueScore (OpenCV 4.10.0 / C++23)**

The Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE) is a state-of-the-art image quality assessment algorithm that quantifies the perceptual quality of images without requiring a reference image ([Mittal et al., IEEE Transactions on Image Processing](#)). OpenCV 4.10.0, optimized with C++23, provides a high-performance computing framework for implementing BRISQUE in real-time applications.

### **Why OpenCV 4.10.0 and C++23 for BRISQUE?**

- **Easy-to-Use .NET API**

The Luxoria.Algorithm.BrisqueScore package provides a .NET wrapper around the BRISQUE algorithm, making it accessible for .NET applications. It enables developers to compute image quality scores without a reference image.

- **Cross-Platform Support**

The package includes precompiled native libraries for x86, x64, and arm64, ensuring compatibility across a wide range of Windows devices.

- **Efficient Image Quality Analysis**

BRISQUE extracts spatial features and computes perceptual quality scores, making it ideal for automated and real-time quality assessment in photography, medical imaging, and security applications ([OpenCV Docs](#)).

- **Optimized for Modern Hardware**

OpenCV 4.10.0 leverages SIMD optimizations, AI-assisted vision processing, and GPU acceleration, offering substantial performance gains over previous versions. C++23 further

improves efficiency with enhanced multi-threading and memory management features ([ISO C++23](#)).

- **Seamless Interoperability with .NET and WinUI 3**

Through P/Invoke, C++/CLI, and WinRT, OpenCV integrates seamlessly with .NET-based applications, allowing for high-performance real-time image processing within WinUI 3 ([Microsoft Interop Guide](#)).

- **Extensive Image Processing Features**

OpenCV provides a comprehensive set of tools for edge detection, noise reduction, object recognition, and image enhancement, essential for automated photography and visual quality control applications ([MIT Computer Vision Lab](#)).

## Sentry SDK

Sentry is an industry-leading real-time error tracking and performance monitoring tool that helps developers detect, diagnose, and fix issues in production environments. It provides deep .NET and C++ integration, making it an essential component for improving the stability and maintainability of WinUI 3 applications.

### Why Sentry SDK for WinUI 3?

- **Automatic Error Tracking**

Sentry captures unhandled exceptions, crashes, and memory leaks in real-time, allowing developers to proactively address issues ([Sentry Docs](#)).

- **Real-Time Alerts & Monitoring**

Supports configurable alerting for slow response times, crashes, and memory leaks, ensuring high application availability.

- **Contextual Debugging**

Sentry logs stack traces, breadcrumbs, and system states, providing developers with the full execution context at the time of failure ([Sentry Debugging Guide](#)).

- **.NET & C++ Integration**

Supports both managed (.NET) and unmanaged (C++) code, making it ideal for hybrid applications that require high-performance computing.

- **Cloud-Based or Self-Hosted Flexibility**

Developers can choose between SaaS-based hosting for scalability or self-hosted solutions for enterprise security.

- **Performance Monitoring & Distributed Tracing**

Tracks slow function executions, API latency, and database queries, ensuring a smooth user experience ([Google Web Performance Guide](#)).

## Conclusion

By integrating SkiaSharp, Luxoria.Algorithm.BrisqueScore (OpenCV 4.10.0 / C++23), and Sentry SDK, we ensure that our WinUI 3 application is highly performant, visually compelling, and robust in production environments.

- **SkiaSharp** enables smooth, high-quality UI rendering, enhancing user experience.
- **Luxoria.Algorithm.BrisqueScore** provides advanced image analysis and processing capabilities for vision-based applications.
- **Sentry SDK** ensures application stability, reliability, and proactive debugging.

This technology stack enables us to build scalable, performant, and visually appealing applications that stand out in both functionality and user experience.