Mobile and embedded computing platforms are increasingly being used for computer vision tasks. Currently, these platforms either do all computation on the CPU or offload the processing to another computer. Present computer vision libraries, such as OpenCV, do not utilize the GPUs on these platforms. For GPU processing, OpenCV requires that the GPU support CUDA or OpenCL. Mobile GPUs do not intrinsically support these languages. However, many mobile GPUs have built-in support for the APIs, OpenCL and OpenGL ES. OpenGL ES is a 3D graphics engine, and OpenVG a 2D vector and raster graphics engine. OpenVG is easy to use and has built-in functionality for photo manipulation that makes it an ideal platform for creating a computer vision system.

Getting OpenVG and OpenGL ES to run, requires EGL. EGL creates the rendering environment. However, getting EGL setup and running correctly is non-trivial and device dependent, which is why CVPI also provides an interface to simplify this process.

OpenVG provides methods for filtering image data.

- **vgConvolve**: vgConvolveSeparableConvolve
- **vgGaussianBlur**: EglConvolveSingle

These functions may operate on single image channels. Each channel can be a 32-bit unsigned pixel or 256. vgConvolve uses a different kernel texture for each channel. vgGaussianBlur uses a single kernel texture. The kernel image is created from the kernel matrix. A 1D filter is applied to each pixel in the input image.

- **vgColorMatrix**: EglConvolveSingle

This function changes the color coordinate system of the pixels in the input image. The input image is converted to the new color system, and the values are mapped back to the original color system.

References