ImageXPro – C++ Sample Codes:

Loading ImageXPro:

```
IImageProcess* m_iMstImageProcess;
CoCreateInstance        (__uuidof(CImageProcess),
                         NULL,
                         CLSCTX_INPROC_SERVER,
                         __uuidof(IImageProcess),
                         reinterpret_cast<void **> (&m_iMstImageProcess)
                         );
unsigned char* pchProcessedBuffer = new unsigned char[ImageSize];
```

Rotation

//For rotating the image at any angle i.e nAngle = 0 to 360.

```
m_iMstImageProcess ->Rotate(pchProcessedBuffer, (int*)pBitmapInfo,
                                        pchImgBuffer, nAngle);
```

Brightness & Contrast

```
m_iMstImageProcess->Intensity([out] int* pProcessedLUT,
                                        [out] unsigned char* pchProcessedBuffer,
                                        [in] int* pBitmapInfo,
                                        [in] unsigned char* pchImgBuffer, int nBright,
                                        int nContrast);
```

Sobel EdgeDetection:

```
m_iMstImageProcess->EdgeDetect(pchProcessedBuffer,
                                        (int*) pBitmapInfo,
                                                pchImgBuffer);
```

Laplacian EdgeDetection:

```
m_iMstImageProcess->LaplacianEdgeDetection(pchProcessedBuffer,
                                                        (int*) pBitmapInfo,
                                                        pchImgBuffer);
```

Zoom:

```
m_iMstImageProcess->Scale(pchProcessedBuffer,
                                (int*) pBitmapInfo,
                                pchImgBuffer,
```

```
                                        nScaledWidth,
                                        nScaledHeight);
```

Smoothness:

```
m_iMstImageProcess->Smooth(pchProcessedBuffer,
                                        (int*) pBitmapInfo,
                                        pchImgBuffer);
```

Sharpness:

```
m_iMstImageProcess->Sharp(pchProcessedBuffer,
                                        (int*) pBitmapInfo,
                                        pchImgBuffer);
```

Emboss:

```
m_iMstImageProcess->Emboss(pchProcessedBuffer,
                                        (int*) pBitmapInfo,
                                        pchImgBuffer);
```

Blur:

```
m_iMstImageProcess->Blur(pchProcessedBuffer,
                                        (int*) pBitmapInfo,
                                        pchImgBuffer);
```

Gamma:

```
m_iMstImageProcess-> Gamma(pchProcessedBuffer,
                                        (int*) pBitmapInfo,
                                        pchImgBuffer);
```