Opencv posee varias funciones con las que podemos crear figuras geométricas o formas más complejas, dibujar textos sobre las imágenes, vamos a ver cuáles son estas funciones y cómo usarlas en nuestros proyectos con opencv.

Lo primero que necesitamos es donde dibujar por lo que crearemos una imagen (Mat) vacía sobre la cual representaremos nuestros dibujos, el siguiente código crea un objeto de la clase Mat que representa nuestra imagen en opencv de 800x600 pixeles, de 3 canales BGR, e inicializada en 0 o sea color negro.

Mat img(rows, cols, CV\_8UC3, Scalar::all(0));

**rows:**cantidad de filas o pixeles de alto que tendrá nuestra imagen 600.

**cols:**cantidad de columnas o pixeles de ancho de nuestra imagen 800.

**CV\_8UC3:** puede ser alguno de los valores mencionados abajo seguido de C3 para tres canales, o C1, C2, C4 para 1, 2, 4, canales respectivamente:

CV\_8U - 8-bit unsigned integers ( 0..255 )

CV\_8S - 8-bit signed integers ( -128..127 )

CV\_16U - 16-bit unsigned integers ( 0..65535 )

CV\_16S - 16-bit signed integers ( -32768..32767 )

CV\_32S - 32-bit signed integers ( -2147483648..2147483647 )

CV\_32F - 32-bit floating-point numbers ( -FLT\_MAX..FLT\_MAX, INF, NAN )

CV\_64F - 64-bit floating-point numbers ( -DBL\_MAX..DBL\_MAX, INF, NAN )

**Scalar::all(0):**Scalar representa el color de la imagen en formato BGR, podemos establecer un color diferente estableciendo los diferentes canales de este modo: Scalar(azul, verde, rojo), podemos usar el macro CV\_RGB(rojo, verde, azul) si nos resulta más cómodo.

Ahora podemos comenzar a dibujar, la primera función que veremos es circle la cual nos sirve para dibujar un circulo, es muy sencilla veamos cómo se usa:

circle(img, Point(cols / 2, rows / 2), 250, Scalar(255,0,0), 3);

Para usar esta función debemos indicar en qué imagen vamos a dibujar, el punto donde se ubicara el centro del circulo seguido del radio y el color, existen otros parámetros de esta función pero son opcionales, en este caso el ultimo 3 indica el grosor de la línea de dibujo.

Además existen la funciones line, rectangle, entre otras cuyo uso es muy similar.

Otra función interesante de dibujo es la función putText la cual dibuja en texto, para la misma debemos especificar la imagen donde vamos a dibujar, el texto a representar, el punto donde será dibujado el texto (la parte izquierda inferior), el tipo de letra (FONT\_HERSHEY\_SIMPLEX, FONT\_HERSHEY\_PLAIN, FONT\_HERSHEY\_DUPLEX, FONT\_HERSHEY\_COMPLEX, FONT\_HERSHEY\_TRIPLEX, FONT\_HERSHEY\_COMPLEX\_SMALL, FONT\_HERSHEY\_SCRIPT\_SIMPLEX, FONT\_HERSHEY\_SCRIPT\_COMPLEX ), el tamaño, color, y grosor del texto.

putText(img, "OpenCV 2", Point(180,320), FONT\_HERSHEY\_SCRIPT\_COMPLEX, 3, CV\_RGB(125,12,145), 2);

Podemos encontrar algunos ejemplos de como dibujar en opencv en la carpeta: **C:\Opencv\samples\cpp\tutorial\_code\core\Matrix**

**Código de ejmplo:**

#include <opencv2\opencv.hpp>

using namespace cv;

void main()

{

       int rows = 600;

       int cols = 800;

       int rec = 150;

       //crear una imagen de 800x600 pixeles

       //canal BGR de tipo CV\_8U( valores de 0 a 255)

       //inicializamos a 0 (color negro)

       Mat img(rows, cols, CV\_8UC3, Scalar::all(0));

       //dibuja un circulo en el centro de la imagen de un radio de 250

       circle(img, Point(cols / 2, rows / 2), 250, Scalar(255,0,0), 3);

       //dibuja una linea

       line(img, Point(), Point(cols, rows), CV\_RGB(255,0,0), 2, CV\_AA);

       //dibuja un rectangulo

       rectangle(img, Point(rec, rec), Point(cols - rec, rows - rec), CV\_RGB(0,255,255));

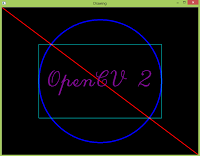
       //dibuja el texto Opencv 2

       putText(img, "OpenCV 2", Point(180,320), FONT\_HERSHEY\_SCRIPT\_COMPLEX, 3,CV\_RGB(125,12,145), 2);

       imshow("Drawing", img);

       waitKey(0);

}

[](http://3.bp.blogspot.com/-V6OaoHPMyLE/UZeHsZ_OSLI/AAAAAAAAAPg/qaxGSXEAKSw/s1600/1.+drawing+opencv.png)