UNIVERSIDAD
COMPLUTENSE
MADRID

OpenCV

Dotando a las máquinas de visión

Participantes:
Alonso Núñez, Mario
Raimundo Fernando, Jose Eduardo

Master en Ingeniería Informática

INDICE DE CONTENIDOS

¿QUÉ ES OPENCV?

OpenCv es una biblioteca libre para el procesamiento de imágenes y video desarrollada inicialmente por Intel.

Se trata de la herramienta más potente dentro del ámbito de la visión artificial y goza de una gran popularidad.

Empleada en multitud de ámbitos como robótica, reconocimiento de objetos, seguridad, impresión 3D y ocio

¿Qué es openCV?

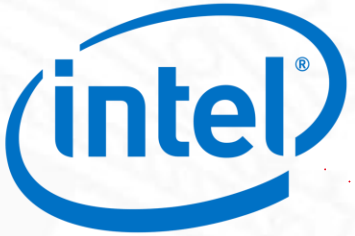
OpenCv nació como un proyecto de computación en altas prestaciones que buscaba abarcar la totalidad del ámbito de la visión artificial.

Su principal objetivo es proporcionar un código optimizado y una infraestructura estable para el procesamiento de imágenes

Abordaba algunos proyectos como el trazado de rayos en tiempo real y la visualización tridimensional.

El desarrollo de la biblioteca empezó en 1999 y su versión 1.0 fue lanzada en 2006 tras 5 años en fase alfa.

Surge una brutal competencia entre Intel y Nvidia por el procesamiento de imágenes en altas prestaciones, enfrentando a la CPU y la GPU en computo paralelo.



Tras una aplastante derrota debido al creciente número de núcleos de la GPU, Intel decide liberalizar OpenCV y trabajar en su nueva versión 2.0

PLANTEAMIENTO DE LOS DATOS

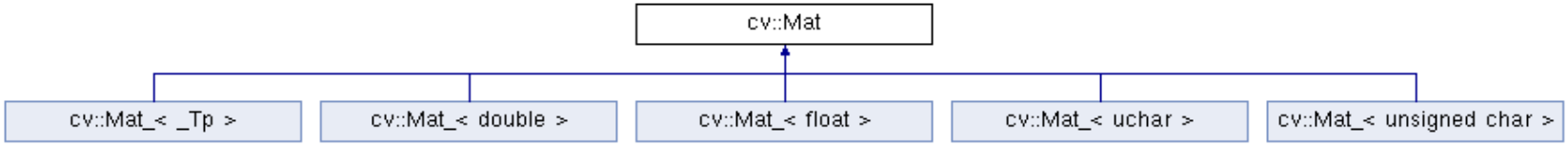
OpenCV da una gran importancia a la homogeneidad de los datos, haciendo que toda la información procesable sean objetos de la misma clase, llamada ***MAT***.

Una imagen se representa como un solo objeto de dicha clase, mientras que un vídeo es un array de objetos.

El gran potencial del tratamiento de la información reside en cómo se configuran los campos internos de dicha clase, lo cual permite una gran versatilidad.

Planteamiento de los datos

La clase ***MAT*** contiene toda la información necesaria para el tratamiento de los datos y podemos usar diferentes subclases según la imagen a procesar.



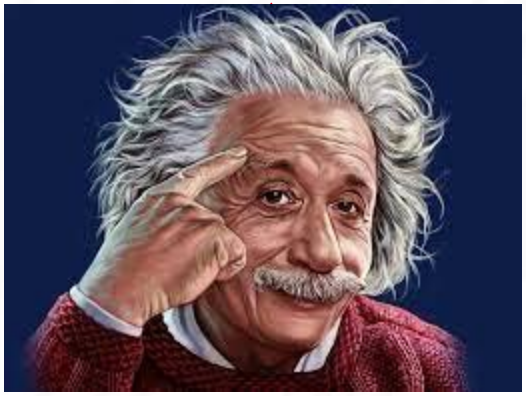
El componente principal de la clase son los elementos ***Vec***, cada uno de los cuales representa una capa de la imagen y se organizan dentro de un array.

Una imagen RGB estará compuesta por un total de tres elementos ***Vec*** en el array, correspondientes con cada una de las diferentes capas de color.

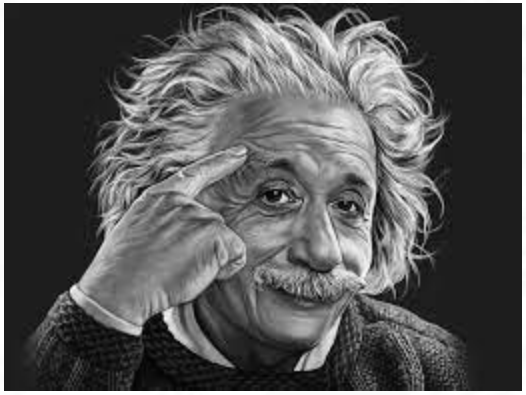
Planteamiento de los datos

Cada uno de los elementos ***Vec*** será una matriz de las mismas dimensiones que la imagen o video, y cada uno de sus elementos tendrá un valor entre 0 y 255.

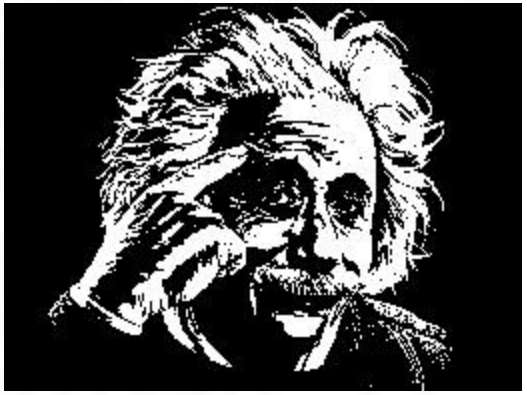
Para controlar el tamaño de la información, un mismo objeto ***MAT*** puede tener un número diferente de matrices ***Vec***, cuyos elementos número ocupan una cantidad específica de bits.



3 matrices Vec de 8 bits



1 matriz Vec de 8 bits

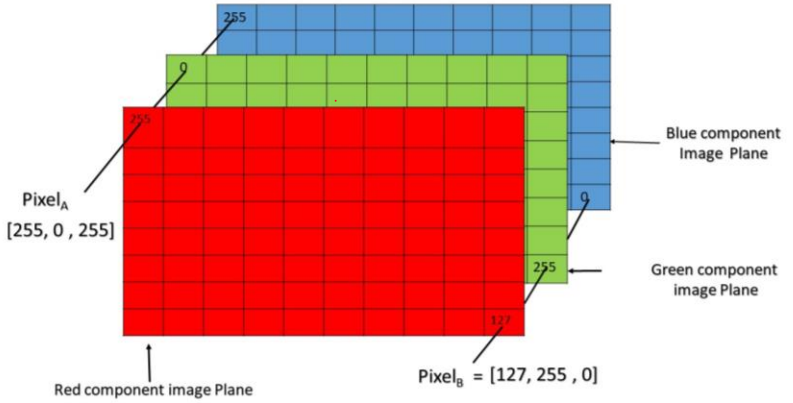


1 matriz Vec de 1 bit

OPERACIONES

OpenCV proporciona una larga lista de operaciones que podemos realizar sobre los objetos de la clase ***MAT***. Diferenciamos tres tipos:

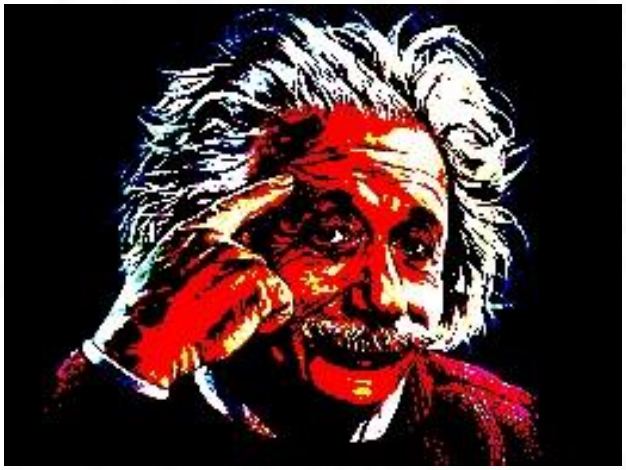
- Operaciones de pixel.
- Operaciones de mallado.
- Operaciones de imagen.



Estos grupos hacen referencia al área de la imagen que tiene que analizarse a la hora de aplicar una única vez dicha operación.

Operaciones de pixel

Aplicadas a cada pixel de la imagen independientemente del resto de pixeles del entorno. Algunas son:



Filtro Threshold

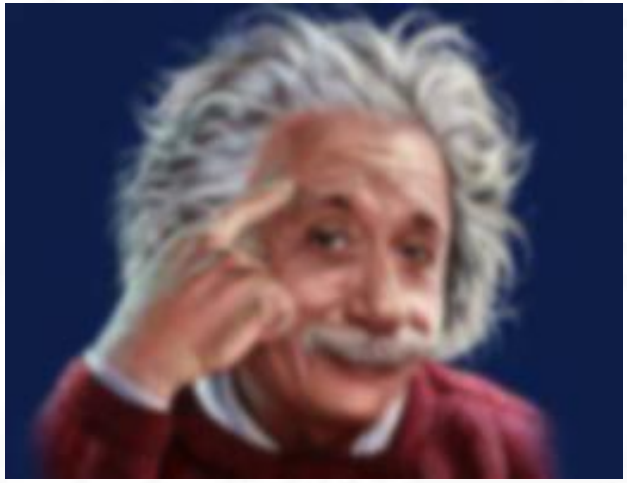
Todo pixel con un valor superior a un umbral es sustituido por un cierto valor dado

Operación pixel a pixel

Acceso a todos los píxeles que componen las matrices ***Vec*** y puesta a cero de todos sus elementos, dejando únicamente una de las matrices sin modificar

Operaciones de mallado

Aplicadas a cada pixel de la imagen dependiendo de los píxeles que lo rodean. Algunas son:



Filtro Gauss

Normaliza el conjunto de elementos del mallado reduciendo la diferencia de los valores extremos.

Filtro Canny

Analiza los elementos del mallado detectando cambios bruscos para identificar los bordes

Operaciones de imagen

El valor resultante de cada pixel puede llegar a depender del valor de toda la imagen Algunas son:

Filtro Sobel

Identifica todas las líneas que forman la imagen y mantiene aquellas que tienen una inclinación específica

Detección de esquinas

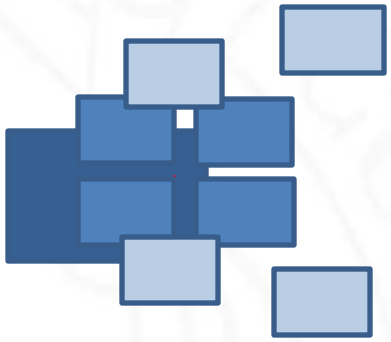
Analiza completamente la imagen para detectar las esquinas de la misma

OPENCV EN AMBITOS DISTRIBUIDOS

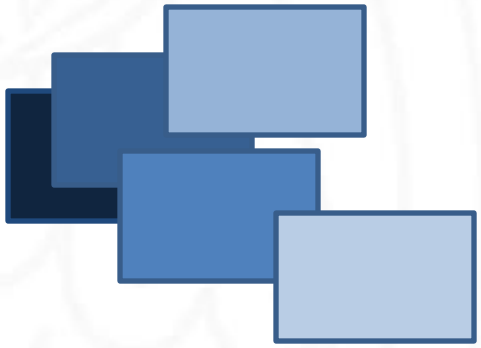
OpenCV está disponible tanto en C++ como en Python, por lo que puede utilizarse con múltiples herramientas de cómputo distribuido como ***MPI*** o ***MrJob***.

El esquema empleado para el procesamiento de los datos dependerá del tamaño de cada uno de los elementos ***MAT*** y de la urgencia de dicho procesamiento.

Si necesitamos procesar imágenes con una gran resolución, dicho ***MAT*** será fragmentado en múltiples objetos ***MAT*** y distribuidos para su procesamiento.



Si necesitamos procesar video en tiempo real, necesitaremos distribuir los objetos ***MAT*** a distintos equipos según estos se van generando.



INTERACCIÓN CON MACHINE LEARNING

Los algoritmos de aprendizaje automático emplean datos de entrenamiento, los cuales incluyen varios componentes.

OpenCv cuenta con el módulo **ml**, el cual asume que todos los vectores de entrenamiento tienen el mismo número de componentes.

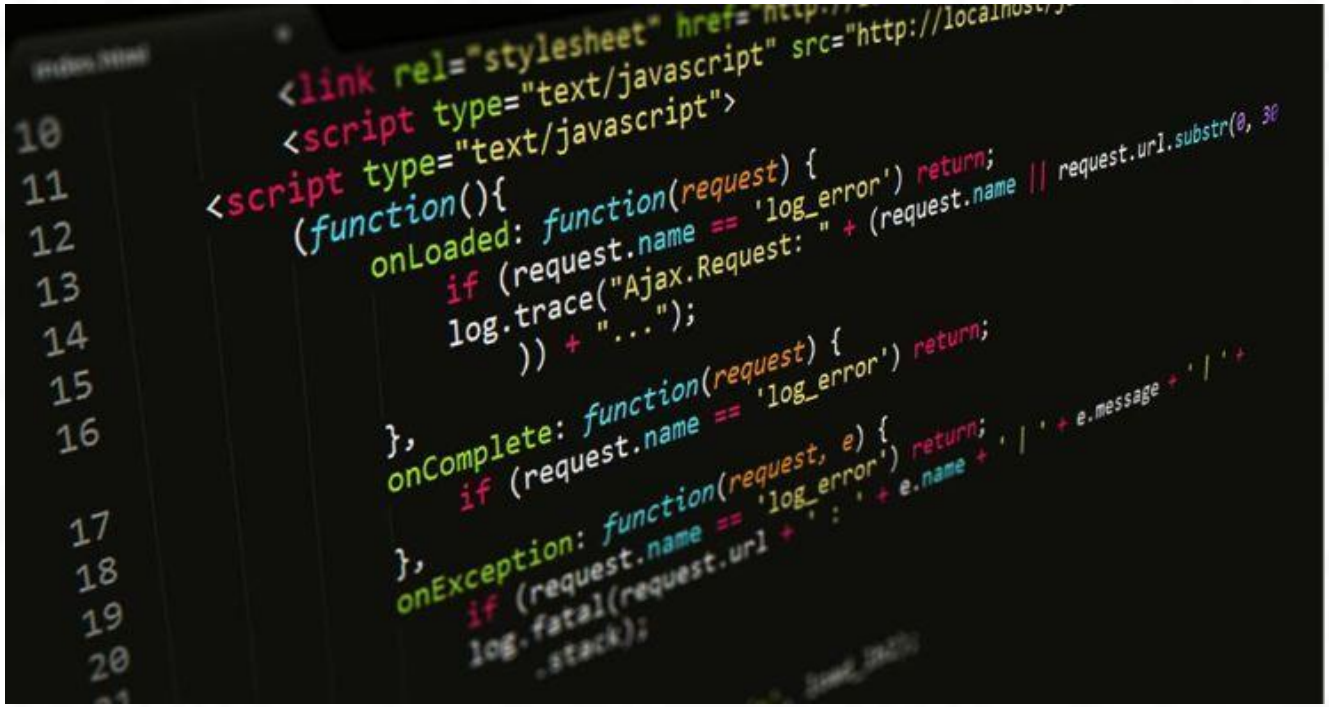
Para hacer una abstracción de los datos de entrenamiento en OpenCV, ml cuenta con la clase **TrainData**.

- *pip install opencv-python*
- *pip install opencv-contrib-python*

Algunas clases del módulo **ml:**

- cv::ml::EM
- cv::ml::Boost
- cv::ml::DTrees
- cv::ml::RTrees
- cv::ml::KNearest
- cv::ml::ANN_MLP
- cv::ml::LogisticRegression
- cv::ml::NormalBayesClassifier

EJEMPLOS


<link rel="stylesheet" href="http://
<script type="text/javascript" src="http://localhost/
<script type="text/javascript">
(function(){
onLoaded: function(request) {
if (request.name == 'log_error') return;
log.trace("Ajax.Request: " + (request.name || request.url.substr(0, 30
)) + "...");
},
onComplete: function(request) {
if (request.name == 'log_error') return;
},
onException: function(request, e) {
if (request.name == 'log_error') return;
log.fatal(request.url + ' : ' + e.name + ' | ' + e.message + ' | ' +
.stack);

BIBLIOGRAFÍA

- Machine Learning
- Documentación clase MAT
- Documento: Informe_openCV_tratamiento_imagenes
- Web: Descripción general del aprendizaje automático
- Youtube: Reconocimento facial