



Destapando superhéroes

Xamarin + OpenCV + Cognitive Services

Sponsors

Sin ellos no sería posible el evento!



MC _18

Yeray Julián



@josueyeray / jyeray@devsdna.com

Microsoft MVP, antes Xamarin MVP y Nokia Developer Champion.

CTO en [devsdna](#), desarrollador móvil desde que puse mis dedos sobre una Sharp Zaurus SL5500

Living la vida @ Tenerife!



Llevando OpenCV a Xamarin

Librería de visión por ordenador en tiempo real open source, lanzada en 1999

Licencia BSD

Reconocimiento facial

Segmentación y reconocimiento de formas

Tratamiento de imágenes, identificación de objetos

Amplio soporte: Windows, Linux, Mac, iOS, Android, C++, Python, Java...



Llevando OpenCV a Xamarin

OpenCV implementación nativa: iOS y Android

Necesitamos comunicarnos con el componente... ¿Cómo?

Recordemos como funciona Xamarin, C# solo en tiempo de desarrollo, tras compilar se comunica con componentes nativos 100%

Podemos crear un binding que nos permitirá interactuar con la implementación nativa desde C#.

Es como un puente, mapeamos toda la superficie pública del componente a C# de forma automática. Al compilar será sustituido por el componente real.



Llevando OpenCV a Xamarin



En iOS, la herramienta **objective sharpie** nos permite leer e interpretar un .h y crear clases de C#

Crearemos un proyecto de binding de iOS con estas clases C# y el archivo binario .a de OpenCV.

La complejidad es directamente proporcional a la superficie del componente a mapear.

OpenCV es un framework muy complejo, con miles de miembros públicos.

Crearemos una librería estática universal en Objective C que use OpenCV y será esta la que mapearemos a C#, solo con los métodos que necesitemos.



devs dna



Llevando OpenCV a Xamarin

En Android, añadiendo el .jar o .aar a un proyecto de Android binding, se genera el código necesario.

Tenemos archivos XML que nos permiten controlar aspectos del binding como:

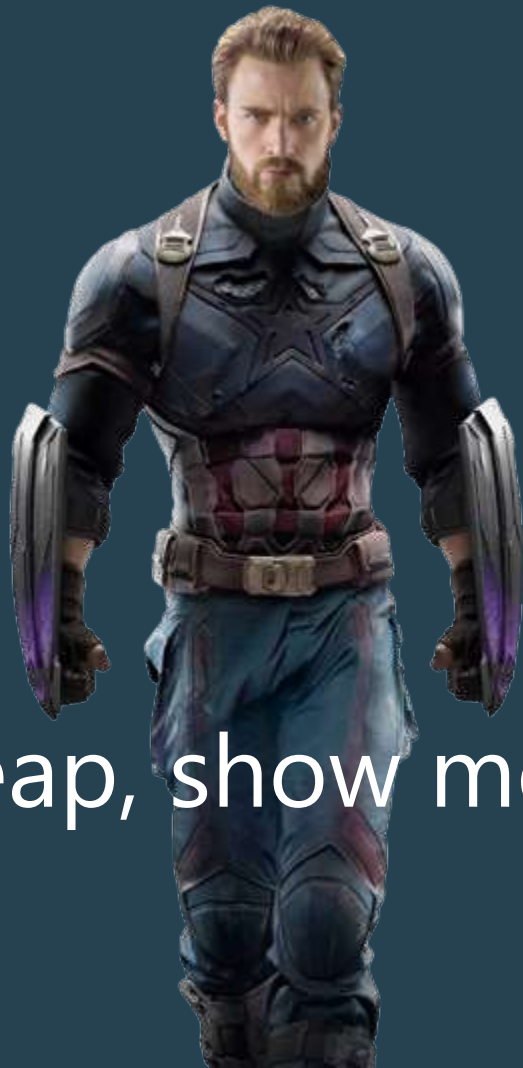
Mapeo de namespaces de Java a C#, para cambiar el formato de dominio inverso a .NET

Mapeo de métodos, para cambiar los nombres o los tipos de parámetros de Java a C#

Definición de estructuras

Al compilar, se copia el .jar o .aar y nuestra app accederá a las librerías nativas.





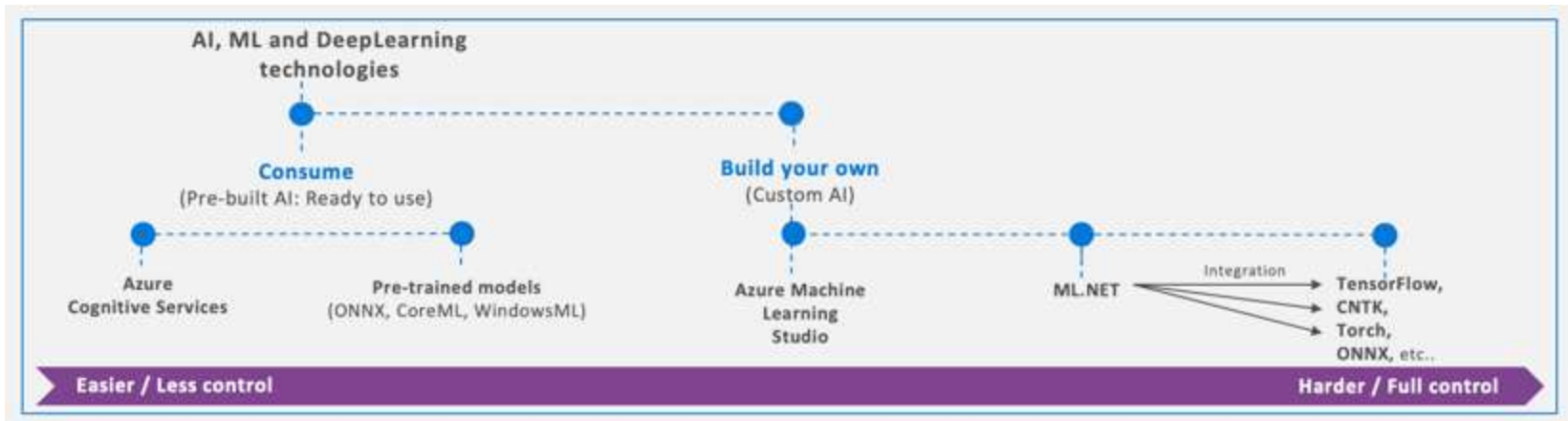
Talk is cheap, show me the code



Cognitive Services

Reconocimiento de caras con Cognitive Services

Azure nos ofrece todo el rango posible de trabajo con Inteligencia Artificial





Cognitive Services

Reconocimiento de caras con Cognitive Services

Machine Learning, Deep Learning son la base, pero requieren de mucha especialización y muchos conocimientos.

Cognitive Services es Inteligencia Artificial como servicio (AlaaS), nos ofrece muchos servicios diferentes:

- Lenguaje: L.U.I.S., OCR, traducción, moderación, análisis...
- Vision: **Clasificación de imágenes**, reconocimiento, OCR, **Custom Vision**...
- Search: Búsquedas contextuales, por imágenes o videos...
- Voz: Verificación de voz, reconocimiento, conversión...
- Conocimiento: Extracción de información desde texto des estructurado...





Cognitive Services

Reconocimiento de caras con Cognitive Services

Cognitive Services tiene un objetivo:

"Democratizar Machine Learning y la IA en general para los desarrolladores"

Cost function

Logistic regression:

$$\underline{J(\theta)} = -\frac{1}{m} \left[\sum_{i=1}^m y^{(i)} \log h_{\theta}(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)})) \right] + \frac{\lambda}{2m} \sum_{j=1}^n \theta_j^2$$





Cognitive Services

Reconocimiento de caras con Cognitive Services

Vamos a usar el reconocimiento de caras de Cognitive Services.

Todas las APIs de Cognitive Services tienen endpoints REST

`https://[location].api.cognitive.microsoft.com/face/v1.0`

Necesitamos indicarle nuestra suscripción y qué información queremos obtener:

- Edad, Genero
- Sonrisa, emociones
- Pose, pelo, gafas, maquillaje...





Talk is cheap, show me the code



CustomVision

Clasificación de imágenes con CustomVision

Un componente más de Cognitive Services.

CustomVision nos permite crear un clasificador de imágenes, alimentado con nuestro propio entrenamiento.

Internamente, es una red neural, a la que se ha añadido un frontal web que nos permite agregar el set de imágenes que queremos usar como base de entrenamiento.



devs dna





CustomVision

Clasificación de imágenes con CustomVision

Dispone de dominios pre entrenados, que podemos extender con nuestra propia biblioteca:

- Adultos
- Comida
- Retail
- Edificios
- General

Algunos dominios son de tipo compacto, pueden descargarse y usarse con CoreML o TensorFlow lite.

Al incluir imágenes, podemos indicarle una o más etiquetas.

Para crear una etiqueta necesitaremos al menos 5 imágenes, antes de poder entrenarlo.

Según lo vayamos usando, podremos ir incorporando las imágenes procesadas a su entrenamiento para mejorar el reconocimiento.



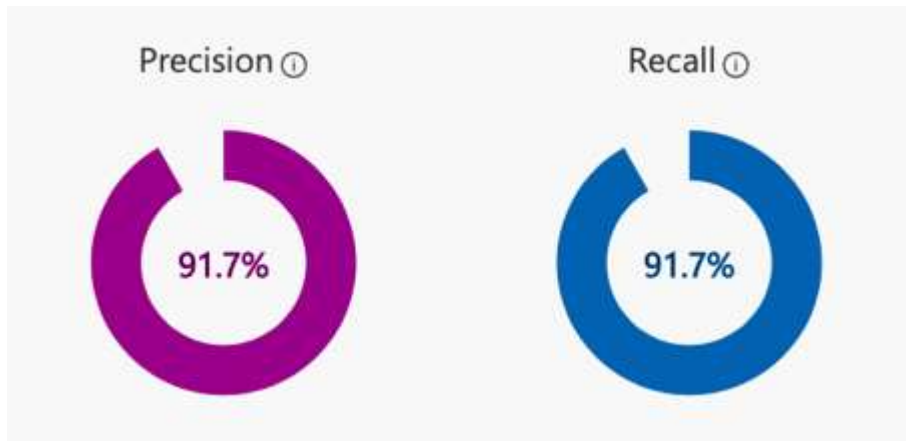


CustomVision

Clasificación de imágenes con CustomVision

Tras entrenar un set de imágenes obtenemos dos indicadores: precisión y recuerdo.

- Precisión indica el % de acierto al reconocer un tag.
- Recuerdo indica el % de probabilidad de reconocer el tag correcto en nuevas imágenes.
- Por regla general, subir el recuerdo disminuye la precisión. Lo mejor es alto recuerdo con una precisión algo más baja. Un 10% de diferencia entre ambos, es óptimo.





Talk is cheap, show me the code

¿Quieres saber más sobre Xamarin? Mira nuestro canal de YouTube: <http://bit.ly/DevsDNA>



Monkey Conf _18

¡GRACIAS!