

# Práctica Imágenes - Detección de objetos

Pablo Mena

Beltrán Rodríguez-Mon

---

Para la elaboración de esta práctica hemos utilizado YOLOv3 para detectar imágenes de Tablas dentro de documentos.

Para realizar el entrenamiento primero de todo hemos inicializado el modelo de YOLO utilizando unos pesos predefinidos basados en la red **Darknet** (red CNN).

\*Dentro del repositorio de Github se ha omitido la carpeta con los datos de Training y con el modelo, para poder subir el resto.

\*\* El repositorio al completo con los datos están compartidos en esta [Carpeta de Drive](#) estando también parcialmente en [Github](#)

## Pipeline

Para construir el modelo YOLO, los pasos seguidos han sido los siguientes:

1. [Anotación de imágenes](#)
  - Tag de las imágenes según (nombre-label-coordenadas)
  - Estos tags se almacenan en un csv dentro de Data/Source\_Images/Training/vott-csv-export
2. [Entrenamiento](#)
  - Descargar los pesos pre-entrenados
  - Entrenar el modelo YOLO custom para la detección de tablas
3. [Inferencia/test](#)
  - Detectar tablas en nuevas imágenes

## Repositorio

- [1\\_Image\\_Annotation](#): Anotación de imágenes
- [2\\_Training](#): Scripts para el entrenamiento del modelo
- [3\\_Inference](#): Scripts para el test
- [Data](#): Input Data, Output Data y Pesos del modelo
- [Utils](#): Scripts de utilidad

## Getting Started

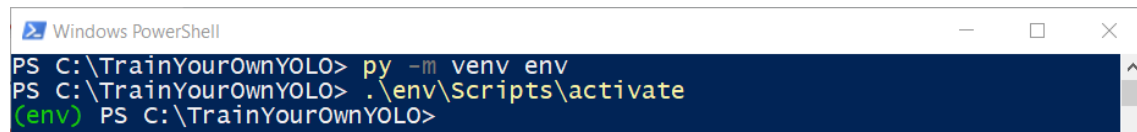
Entorno Virtual (**Linux/Mac**):

```
python3 -m venv env
source env/bin/activate
```

Entorno Virtual (**Windows**):

```
py -m venv env
.\env\Scripts\activate
```

PowerShell



```
Windows PowerShell
PS C:\TrainYourOwnYOLO> py -m venv env
PS C:\TrainYourOwnYOLO> .\env\Scripts\activate
(env) PS C:\TrainYourOwnYOLO>
```

Instalación de los paquetes necesarios [Windows, Mac or Linux]

```
pip install -r requirements.txt
```

Si falla, puede ser que sea necesario ejecutar: `pip install pip --upgrade`.

## Entrenamiento del modelo

El entrenamiento del modelo se ha realizado haciendo *Fine-Tuning* del modelo original. Los resultados han sido

```
45/45 [=====] - 622s 14s/step - loss:
2072.1919 - val_loss: 270.7197
Epoch 2/51
45/45 [=====] - 517s 11s/step - loss:
189.7256 - val_loss: 134.2318
Epoch 3/51
45/45 [=====] - 516s 11s/step - loss:
110.2705 - val_loss: 95.2559
Epoch 4/51
45/45 [=====] - 535s 12s/step - loss: 78.6412
- val_loss: 68.6279
Epoch 5/51
45/45 [=====] - 620s 14s/step - loss: 61.8053
- val_loss: 55.1787
Epoch 6/51
45/45 [=====] - 649s 14s/step - loss: 50.5018
- val_loss: 45.4948
Epoch 7/51
45/45 [=====] - 676s 15s/step - loss: 43.4575
- val_loss: 39.7827
Epoch 8/51
45/45 [=====] - 674s 15s/step - loss: 38.1779
- val_loss: 36.2439
Epoch 9/51
45/45 [=====] - 647s 14s/step - loss: 34.1646
- val_loss: 32.1615
Epoch 10/51
45/45 [=====] - 681s 15s/step - loss: 31.5358
- val_loss: 30.2518
Epoch 11/51
45/45 [=====] - 625s 14s/step - loss: 28.8158
- val_loss: 27.2826
Epoch 12/51
45/45 [=====] - 629s 14s/step - loss: 27.4354
- val_loss: 26.2810
```

Epoch 13/51  
45/45 [=====] - 553s 12s/step - loss: 25.5320  
- val\_loss: 24.6966  
Epoch 14/51  
45/45 [=====] - 550s 12s/step - loss: 24.4519  
- val\_loss: 23.3203  
Epoch 15/51  
45/45 [=====] - 615s 14s/step - loss: 23.3815  
- val\_loss: 22.2603  
Epoch 16/51  
45/45 [=====] - 625s 14s/step - loss: 22.4803  
- val\_loss: 21.5432  
Epoch 17/51  
45/45 [=====] - 632s 14s/step - loss: 21.6809  
- val\_loss: 20.7737  
Epoch 18/51  
45/45 [=====] - 622s 14s/step - loss: 20.9248  
- val\_loss: 21.1336  
Epoch 19/51  
45/45 [=====] - 640s 14s/step - loss: 20.3810  
- val\_loss: 19.3579  
Epoch 20/51  
45/45 [=====] - 646s 14s/step - loss: 19.8482  
- val\_loss: 19.5318  
Epoch 21/51  
45/45 [=====] - 635s 14s/step - loss: 19.3444  
- val\_loss: 18.9209  
Epoch 22/51  
45/45 [=====] - 591s 13s/step - loss: 18.8952  
- val\_loss: 18.6198  
Epoch 23/51  
45/45 [=====] - 545s 12s/step - loss: 18.7238  
- val\_loss: 18.3770  
Epoch 24/51  
45/45 [=====] - 525s 12s/step - loss: 18.2791  
- val\_loss: 17.8375  
Epoch 25/51  
45/45 [=====] - 509s 11s/step - loss: 18.0045  
- val\_loss: 17.1076  
Epoch 26/51  
45/45 [=====] - 506s 11s/step - loss: 17.6227  
- val\_loss: 17.2048  
Epoch 27/51  
45/45 [=====] - 510s 11s/step - loss: 17.3180  
- val\_loss: 16.5084  
Epoch 28/51  
45/45 [=====] - 513s 11s/step - loss: 17.2350  
- val\_loss: 16.1463  
Epoch 29/51  
45/45 [=====] - 514s 11s/step - loss: 16.9569  
- val\_loss: 16.8566  
Epoch 30/51  
45/45 [=====] - 516s 11s/step - loss: 16.8506  
- val\_loss: 15.9374

Epoch 31/51  
45/45 [=====] - 525s 12s/step - loss: 16.6143  
- val\_loss: 16.0619  
Epoch 32/51  
45/45 [=====] - 523s 12s/step - loss: 16.3675  
- val\_loss: 15.2876  
Epoch 33/51  
45/45 [=====] - 514s 11s/step - loss: 16.2447  
- val\_loss: 15.9445  
Epoch 34/51  
45/45 [=====] - 581s 13s/step - loss: 16.1473  
- val\_loss: 14.9769  
Epoch 35/51  
45/45 [=====] - 478s 11s/step - loss: 15.9140  
- val\_loss: 15.4020  
Epoch 36/51  
45/45 [=====] - 471s 10s/step - loss: 15.8976  
- val\_loss: 15.4180  
Epoch 37/51  
45/45 [=====] - 472s 10s/step - loss: 15.7109  
- val\_loss: 15.4537  
Epoch 38/51  
45/45 [=====] - 472s 10s/step - loss: 15.7216  
- val\_loss: 15.3927  
Epoch 39/51  
45/45 [=====] - 471s 10s/step - loss: 15.5720  
- val\_loss: 15.3672  
Epoch 40/51  
45/45 [=====] - 473s 11s/step - loss: 15.4707  
- val\_loss: 14.6734  
Epoch 41/51  
45/45 [=====] - 472s 10s/step - loss: 15.3267  
- val\_loss: 14.9112  
Epoch 42/51  
45/45 [=====] - 471s 10s/step - loss: 15.2894  
- val\_loss: 14.7180  
Epoch 43/51  
45/45 [=====] - 471s 10s/step - loss: 15.2245  
- val\_loss: 14.7386  
Epoch 44/51  
45/45 [=====] - 474s 11s/step - loss: 15.1195  
- val\_loss: 14.4612  
Epoch 45/51  
45/45 [=====] - 473s 11s/step - loss: 15.0545  
- val\_loss: 14.4236  
Epoch 46/51  
45/45 [=====] - 472s 10s/step - loss: 15.0266  
- val\_loss: 14.6731  
Epoch 47/51  
45/45 [=====] - 472s 10s/step - loss: 14.9318  
- val\_loss: 14.3230  
Epoch 48/51  
45/45 [=====] - 472s 10s/step - loss: 14.8995  
- val\_loss: 14.3872

```
Epoch 49/51
45/45 [=====] - 472s 10s/step - loss: 14.8085
- val_loss: 14.6400
Epoch 50/51
45/45 [=====] - 471s 10s/step - loss: 14.7438
- val_loss: 14.6279
Epoch 51/51
45/45 [=====] - 474s 11s/step - loss: 14.7772
- val_loss: 14.2174
```

## Resultados

Las salidas del modelo están almacenadas en [Data/Source\\_Images/Test\\_Image\\_Detection\\_Results](#). Dentro de este directorio hay varias imágenes para mostrar la efectividad del modelo.

## COTIZACIÓN CONDICIONADA

## 2. INFORMACIONES SOBRE EL OBJETO DE COBERTURA

table 0.93

OBJETO DEL SEGURO (Según información facilitada por el Tomador)	
<b>Descripción Actividad</b>	Propietarios de platós e instalaciones de TV alquiladas a terceros: La «Ciudad del audiovisual» Fase I. Las instalaciones de la actividad asegurada se ubican únicamente en la situación de riesgo descrita.
<b>Domicilio</b>	AV ESPAÑA I
<b>C.P. y Población</b>	28760 TRES CANTOS

**LÍMITES DE SUMA ASEGURADOS** (Importe en euros)

<b>Límite máximo por víctima</b>	<b>300.000,00</b>
----------------------------------	-------------------

Representa el máximo que el Asegurador se compromete a pagar por la suma de todas las indemnizaciones, intereses y gastos por cada persona física afectada por un siniestro amparado de daños corporales y los perjuicios consecutivos a los mismos, junto con las que, en su caso, pudieran corresponder a sus causahabientes o perjudicados.

<b>Límite máximo por siniestro</b>	<b>6.000.000,00</b>
------------------------------------	---------------------

Representa el máximo que el Asegurador se compromete a pagar, por la suma de todas las indemnizaciones, intereses y gastos correspondientes a un siniestro amparado por la póliza, con independencia del número de los riesgos afectados y del número de víctimas o perjudicados existentes.

Se considerará que constituye un solo y único siniestro el acontecimiento o serie de acontecimientos dañosos debidos a una misma causa original con independencia del número de reclamantes o reclamaciones formuladas y el número de personas cuya responsabilidad resulte implicada.

<b>Límite máximo por anualidad/período de seguro</b>	<b>6.000.000,00</b>
--	---------------------

Representa el máximo que el Asegurador se compromete a pagar por la suma de todas las indemnizaciones cubiertas, intereses y gastos correspondientes a una misma anualidad de seguro, con independencia de que dichos daños sean imputables a uno o varios siniestros.

El límite agregado máximo de indemnización por anualidad de seguro se verá reducido en su cuantía, en el orden cronológico de presentación de las reclamaciones formuladas por los perjudicados al Asegurado o al Asegurador, por el importe correspondiente de las reservas de siniestros efectuadas, de las indemnizaciones de daños realizadas y de cualquier transacción amistosa o judicial de indemnización aceptada por el Asegurador. A los efectos se entiende por año de seguro el período que media entre la fecha de efecto y la del primer vencimiento, o bien entre dos vencimientos anuales sucesivos o entre el último vencimiento anual y la extinción o cancelación del seguro.

Los límites de indemnización anteriores son aplicables a todos los riesgos que figuren como contratados, salvo que de forma expresa se haya establecido un sublímite de indemnización específico y distinto para alguno de los que se relacionan a continuación:

table 0.97

RIESGOS DE RESPONSABILIDAD CIVIL ASEGURABLES Y SUBLÍMITES ASEGURADOS ESPECÍFICOS		(importe en euros)	FRANQUICIAS ESPECÍFICAS
Riesgo de Explotación	Incluido		300 €
Riesgo de responsabilidad civil frente al arrendatario/inquilino.	Incluido		300 €
Riesgo de accidentes laborales (R.C. Patronal)	Incluido		300 €
PRESTACIONES GARANTIZADAS		(importe en euros)	FRANQUICIAS ESPECÍFICAS
Defensa Responsabilidad Civil	Contratada		
- Límite gastos en caso de conflicto de intereses	1.500,00		
Defensa Responsabilidad Penal	Contratada		
- Límite gastos en caso de conflicto de intereses	1.500,00		

**ANTECEDENTES TENIDOS EN CUENTA PARA CALCULAR LA PRIMA Y TÉRMINOS DE ACEPTACIÓN DEL SEGURO****BASES DE CÁLCULO DE LA PRIMA DEL SEGURO DECLARADAS PARA CONTRATAR**

(due to the larger number of defaults in sample), so these values are maintained.

**table 1.00** Default Frequency and Volatility

	$\bar{p}_\zeta$	$E[1/\hat{n}_\zeta]$	$\sqrt{V[\hat{p}_\zeta]}/\bar{p}_\zeta$	$\sqrt{V[p_\zeta(x)]}/\bar{p}_\zeta$	$\sqrt{V_\zeta(x)}/\bar{p}_\zeta$
AAA	0	0.0092	.	.	1.4
AA	0	0.0030	.	.	1.4
A	0.0005	0.0017	2.4857	1.5896	1.2
BBB	0.0018	0.0026	1.2477	0.3427	0.4
BB	0.0091	0.0038	1.2820	1.1108	1.1
B	0.0474	0.0041	0.6184	0.5492	0.55
CCC	0.1800	0.0360	0.5510	0.2045	0.4

## C Proof of Proposition 1

Let  $y_1$  and  $y_2$  be the CreditMetrics latent variables for two grade  $\zeta$  obligors. Assume that there is only one systematic risk factor and that the two obligors have the same weight  $w_\zeta$  on that risk factor. Thus,

$$\begin{aligned} y_1 &= xw_\zeta + \sqrt{1 - w_\zeta^2}\epsilon_1 \\ y_2 &= xw_\zeta + \sqrt{1 - w_\zeta^2}\epsilon_2. \end{aligned}$$

Conditional on  $x$ , default events for these obligors are independent, so

$$\Pr(y_1 < C_\zeta \& y_2 < C_\zeta | x) = \Pr(y_1 < C_\zeta | x) \Pr(y_2 < C_\zeta | x) = \Phi\left((C_\zeta - xw_\zeta)/\sqrt{1 - w_\zeta^2}\right)^2 = p_\zeta(x)^2.$$

Therefore,

$$\text{Var}[p_\zeta(x)] = E[p_\zeta(x)^2] - E[p_\zeta(x)]^2 = E[\Pr(y_1 < C_\zeta \& y_2 < C_\zeta | x)] - E[p_\zeta(x)]^2.$$

Since  $y_1$  and  $y_2$  each have mean zero and variance one, and have correlation  $w_\zeta^2$ , the unconditional expectation  $E[\Pr(y_1 < C_\zeta \& y_2 < C_\zeta | x)]$  is given by  $\text{BIVNOR}(C_\zeta, C_\zeta, w_\zeta^2)$ . This gives

$$\text{Var}[p_\zeta(x)] = \text{BIVNOR}(C_\zeta, C_\zeta, w_\zeta^2) - \bar{p}_\zeta^2,$$

as required.

**table 1.00** Table 1.00, Central Growth Zone, moderate (default) survival rate.

		A	B	C	D	E	F	G	H
		Years After Planting							
		1-5	6-10	11-15	16-20	21-25	26-30	31-35	36-40
1	Shade Cooling	0.07	0.21	0.33	0.42	0.49	0.51	0.53	0.53
2	Shade Heating	0.19	0.31	0.42	0.47	0.51	0.52	0.53	0.52
3	Climate Cooling and Heating	0.02	0.08	0.18	0.28	0.36	0.42	0.47	0.49
4	Sequestration	0.08	0.28	0.48	0.58	0.58	0.51	0.44	0.35
5	Decomposition	0.0006	0.0005	0.0013	0.0025	0.0038	0.0052	0.0064	0.0075
6	Maintenance	0.09	0.22	0.33	0.41	0.46	0.49	0.50	0.49
7	Production/Program	0.75	0.00	0.00	0.00	0.00	0.00	0.00	0.00

**table 1.00** Table 1.00, Central Growth Zone, high survival rate.

		A	B	C	D	E	F	G	H
		Years After Planting							
		1-5	6-10	11-15	16-20	21-25	26-30	31-35	36-40
1	Shade Cooling	0.07	0.24	0.40	0.51	0.62	0.66	0.70	0.74
2	Shade Heating	0.22	0.36	0.50	0.58	0.64	0.67	0.70	0.72
3	Climate Cooling and Heating	0.02	0.10	0.21	0.34	0.45	0.54	0.62	0.67
4	Sequestration	0.09	0.33	0.57	0.71	0.72	0.66	0.58	0.48
5	Decomposition	0.0003	0.0003	0.0009	0.0017	0.0026	0.0035	0.0043	0.0050
6	Maintenance	0.10	0.26	0.39	0.50	0.58	0.62	0.66	0.67
7	Production/Program	0.85	0.00	0.00	0.00	0.00	0.00	0.00	0.00

**table 1.00** Table 1.00, Central Growth Zone, low survival rate.

		A	B	C	D	E	F	G	H
		Years After Planting							
		1-5	6-10	11-15	16-20	21-25	26-30	31-35	36-40
1	Shade Cooling	0.06	0.17	0.27	0.33	0.37	0.36	0.35	0.33
2	Shade Heating	0.17	0.26	0.34	0.37	0.39	0.37	0.35	0.32
3	Climate Cooling and Heating	0.01	0.07	0.15	0.22	0.27	0.30	0.31	0.30
4	Sequestration	0.07	0.24	0.39	0.45	0.43	0.37	0.29	0.22
5	Decomposition	0.0008	0.0006	0.0017	0.0033	0.0051	0.0069	0.0085	0.0100
6	Maintenance	0.08	0.19	0.27	0.32	0.35	0.35	0.33	0.30
7	Production/Program	0.65	0.00	0.00	0.00	0.00	0.00	0.00	0.00

## Testing del Detector

Para detectar nuevas tablas es necesario ejecutar el Script de [3\\_Inference](#):

```
python Detector.py
```

La salida se almacena en [Data/Source\\_Images/Test\\_Image\\_Detection\\_Results](#)

## Conclusión

El modelo funciona bastante bien, especialmente considerando que el entrenamiento del modelo original (Darknet) no se ha realizado con un dataset mínimamente semejante al objetivo de nuestro enfoque (identificar tablas en documentos).