

Clasificación de defectos en placas de circuitos impresos

Isabella Callejas Mandon - 2202030

Angel Fernando Ortega García - 2211583

Marlon Barajas Acelas - 2202042

Contenido

01

Problema

¿Por qué es útil nuestro proyecto?

02

Objetivo

¿Qué esperamos obtener?

03

Dataset del proyecto

¿Cómo preparamos nuestros datos?

04

Resultados

¿Cómo nos fue finalmente?

05

Conclusiones

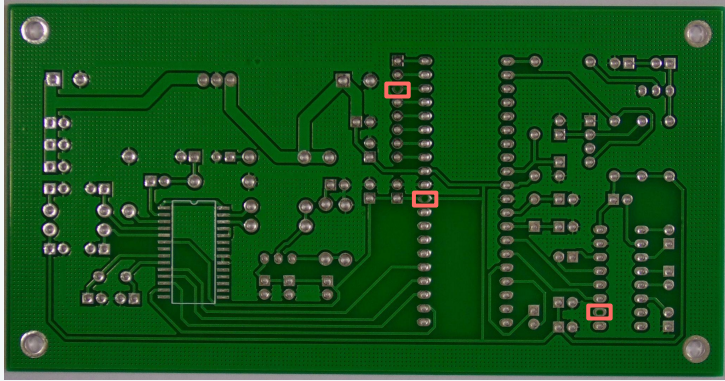
01

Justificación

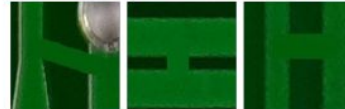
¿Por qué es necesario nuestro proyecto?



Justificación



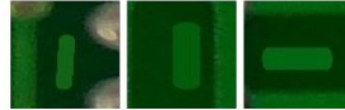
Short



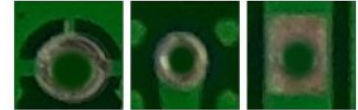
Spur



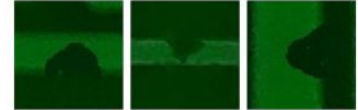
Spurious copper



Missing hole



Mouse bite



Open circuit



Printed Circuit Board – PCB

Las tarjetas electrónicas suelen ser el núcleo de la mayoría de los productos tecnológicos. Básicamente es un dispositivo dónde se instalan componentes electrónicos que se conectan entre ellos [1].

Por lo que, es necesario desarrollar sistemas que puedan ayudar al control de calidad (Automated Optical Inspection) y producción de estas placas [2].

02

Objetivo

¿Qué esperamos obtener?



Objetivo



**Clasificar defectos en placas de
circuitos impresos.**

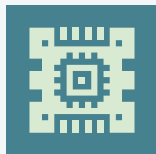
03

Dataset del proyecto

¿Cuáles fueron nuestros datos?



Datasets



PCB Defect Dataset

1386 imágenes, 2953 objetos etiquetados, 6 clases

Missing hole



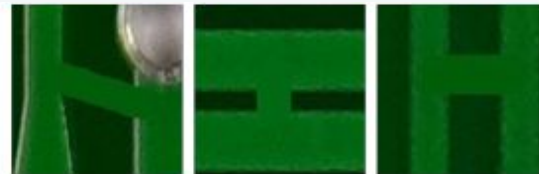
Mouse bite



Open circuit



Short



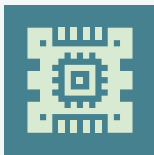
Spur



Spurious copper



Datasets



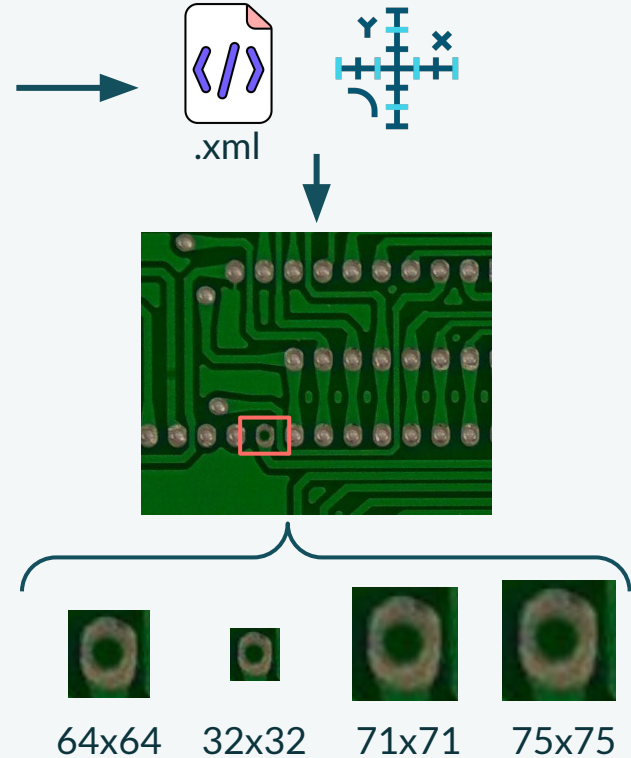
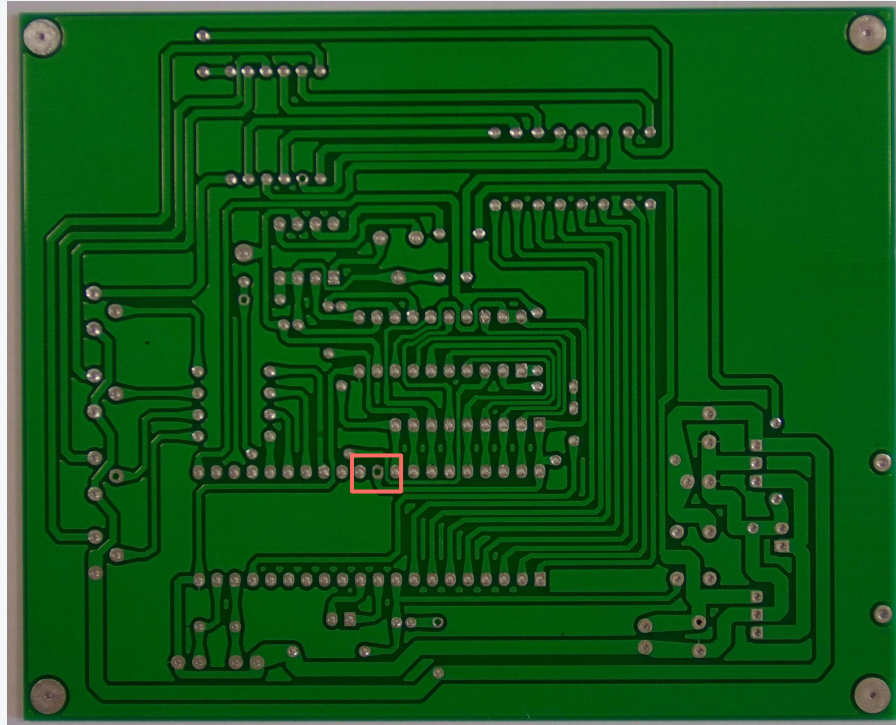
PCB Defect Dataset

1386 imágenes, 2953 objetos etiquetados, 6 clases

Defect	Image number
Mouse_bite	116
Missing_hole	116
Spurious_copper	116
Spur	115
Open_circuit	115
Short	115

Preprocesamiento

Preprocesamiento del dataset



Preprocesamiento

División del dataset

Clase	Train - 64%	Validation - 16%	Test - 20%
Mouse bite	315	79	98
Missing hole	318	79	100
Short	314	79	98
Open_Circuit	309	77	96
Spur	312	79	98
Spurious_copper	321	81	101
Total	1889	474	591

2954 parches!

Resultados

¿Por qué es mejor la Inception V3?

InceptionV3

```
# @title INCEPTION V3
def create_InceptionV3():
    # Cargar el modelo Inception V3 preentrenado
    base_model = InceptionV3(weights='imagenet', include_top=False, input_shape=(75, 75, 3))

    # Añadir capas de clasificación personalizadas
    model = base_model.output
    model = GlobalAveragePooling2D()(model)
    model = Dense(1024, activation='relu')(model)
    predictions = Dense(6, activation='softmax')(model)

    # Crear el modelo completo
    model = Model(inputs=base_model.input, outputs=predictions)

    # Compilar el modelo
    model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

    model.summary()
    return model
```

Resultados preliminares

InceptionV3 – Mejor modelo

Layer (type)	Output Shape	Param #	Connected to
input_layer_2 (InputLayer)	(None, 75, 75, 3)	0	-
conv2d (Conv2D)	(None, 37, 37, 32)	864	input_layer_2[0][0]
batch_normalization (BatchNormalization)	(None, 37, 37, 32)	96	conv2d[0][0]
activation (Activation)	(None, 37, 37, 32)	0	batch_normalization[0...]
conv2d_1 (Conv2D)	(None, 35, 35, 32)	9,216	activation[0][0]
batch_normalization_1 (BatchNormalization)	(None, 35, 35, 32)	96	conv2d_1[0][0]
activation_1 (Activation)	(None, 35, 35, 32)	0	batch_normalization_1...
conv2d_2 (Conv2D)	(None, 35, 35, 64)	18,432	activation_1[0][0]
batch_normalization_2 (BatchNormalization)	(None, 35, 35, 64)	192	conv2d_2[0][0]
activation_93 (Activation)	(None, 1, 1, 192)	0	batch_normalization_9...
mixed10 (Concatenate)	(None, 1, 1, 2048)	0	activation_85[0][0], mixed9_1[0][0], concatenate_1[0][0], activation_93[0][0]
global_average_pooling2d... (GlobalAveragePooling2D)	(None, 2048)	0	mixed10[0][0]
dense_2 (Dense)	(None, 1024)	2,098,176	global_average_poolin...
dense_3 (Dense)	(None, 6)	6,150	dense_2[0][0]

	precision	recall	f1-score	support
Missing_hole	1.00	0.99	0.99	100
Mouse_bite	0.96	1.00	0.98	98
Open_circuit	1.00	0.97	0.98	96
Short	1.00	0.96	0.98	98
Spur	0.96	1.00	0.98	98
Spurious_copper	0.99	0.99	0.99	101
accuracy			0.98	591
macro avg	0.99	0.98	0.98	591
weighted avg	0.99	0.98	0.98	591

Accuracy para la clase 'Missing_hole': 1.00
 Accuracy para la clase 'Mouse_bite': 0.97
 Accuracy para la clase 'Open_circuit': 0.99
 Accuracy para la clase 'Short': 1.00
 Accuracy para la clase 'Spur': 0.97
 Accuracy para la clase 'Spurious_copper': 1.00

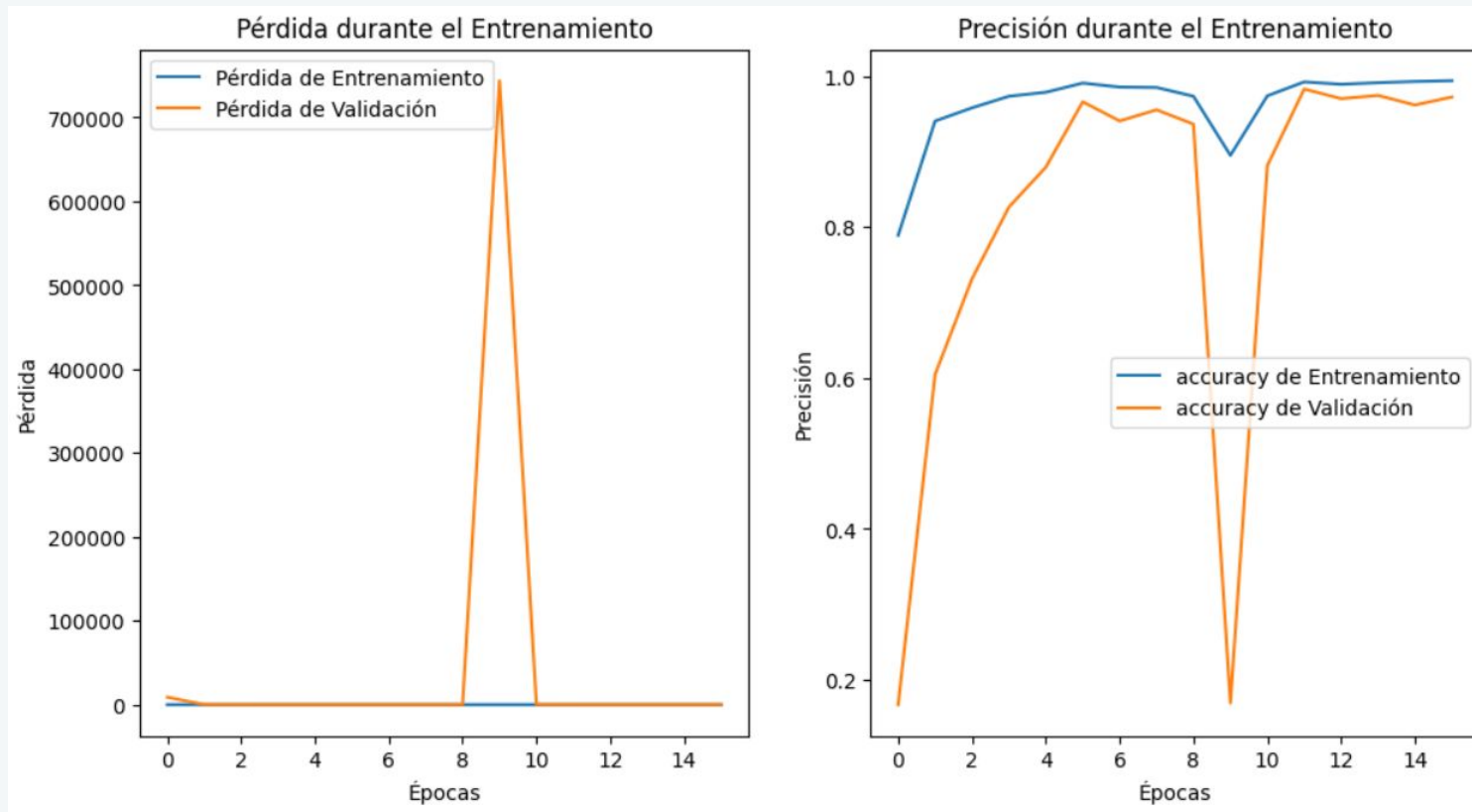
Epoch: 16

Batch size: 32

Learning rate: 0.001

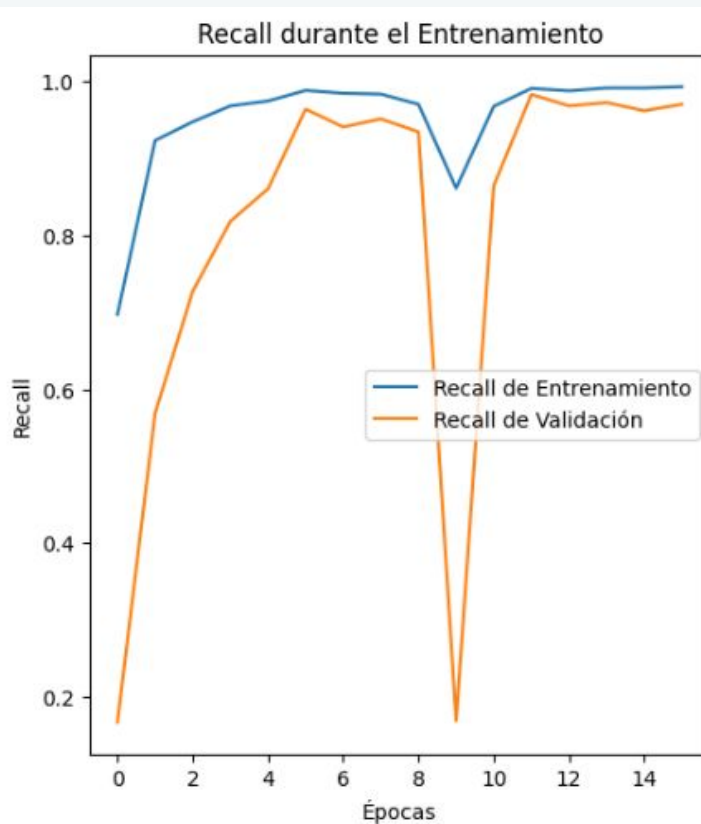
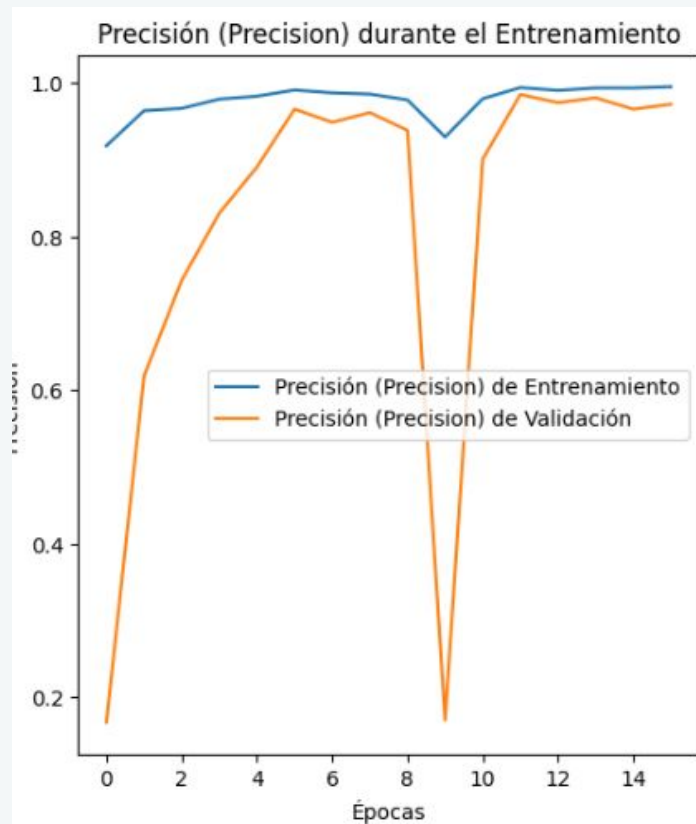
Resultados preliminares

InceptionV3- Mejor modelo



Resultados preliminares

InceptionV3- Mejor modelo



Resultados

Pruebas/variaciones

Cambios	Accuracy	Precision	Recall	AUC
Original	98.82%	98.98%	98.65%	99.89%
Epochs = 32	93.74%	93.73%	93.57%	96.94%
batch size =16	97.46%	97.78%	96.95%	99.84%
More layers	97.63%	97.79%	97.46%	99.87%

04

Resultados

¿Qué obtuvimos finalmente?



Resultados

Estado del arte en precisión (Autor 1)

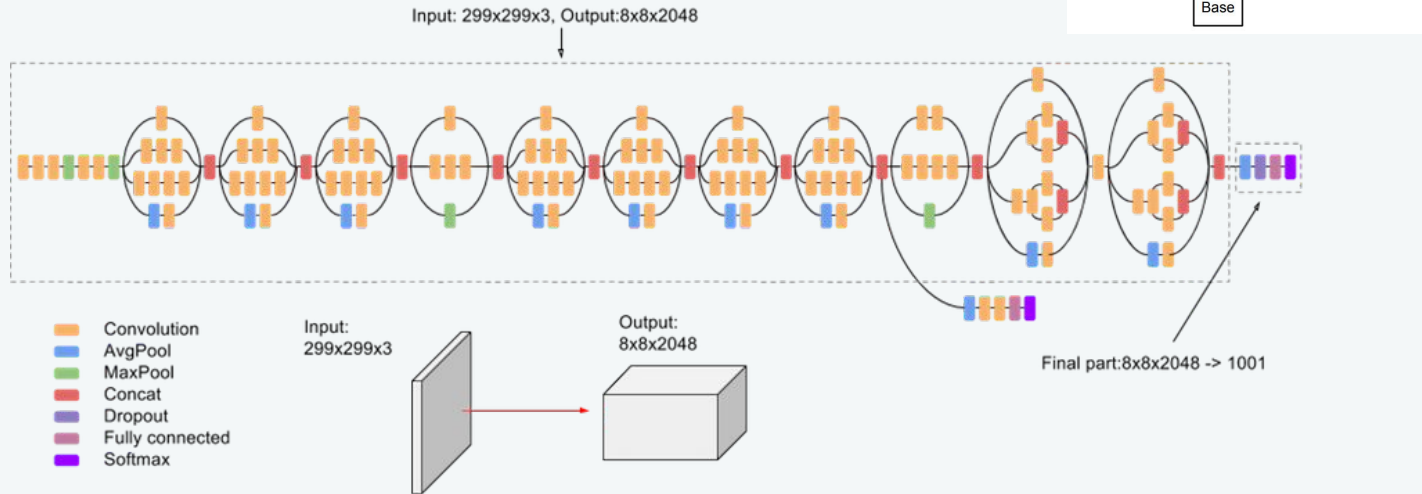
Modelo	Missing Hole	Mouse Bite	Open Circuit	Short	Spur	Spurious Copper
Precision (Autor 1)	98.96%	97.94%	97.74%	99.48%	93.65%	98.52%
DNN	96%	77%	84%	90%	89%	83%
CNN	100%	83%	89%	89%	90%	89%
LeNet	98%	88%	90%	96%	93%	91%
AlexNet	99%	57%	100%	87%	91%	72%
VGG16	99%	90%	95%	91%	94%	87%
ResNet	79%	55%	42%	41%	83%	27%
GoogLeNet	99%	60%	76%	66%	78%	65%
Inception V3	100%	99%	99%	99%	100%	96%
Inception ResNet V2	98%	83%	94%	97%	78%	85%
Xception	94%	94%	99%	92%	100%	92%
MobileNet	100%	85%	83%	81%	84%	92%
DenseNet	100%	90%	99%	92%	87%	91%

Resultados

Arquitecturas de IA2 vs Estado del arte

Arquitecturas	Accuracy	Precision	Recall	AUC	LOSS
Autor 2	89.89% / 92.86%	-----	-----	-----	-----
Autor 3	92.86%	99.8%	99.32%	-----	-----
DNN	86.29%	87.81%	84.09%	97.73%	0.4594
CNN	89.68%	90%	89.83%	99.05%	0.3066
LeNet	92.55%	92.84%	92.22%	99.10%	-----
AlexNet	90.19%	90.17%	90.02%	96.35%	0.7044
VGG16	92.72%	94.46%	92.39	99.18%	0.2262
ResNet50	46.02%	79.71%	18.61%	82.78%	-----
GoogleLeNet	88.66%	88.79%	88.49%	98.24%	0.3932
InceptionV3	98.82%	98.98%	98.65%	99.89%	0.0587
Inception-ResNet-V2	89.17%	90.16%	88.32%	96.94%	-----
Xception	96.11%	96.11%	96.11%	96.16%	0.1852
MobileNet	87.48%	88.01%	86.97%	97.18%	0.5436
DenseNet121	93.23%	94.31%	92.55%	99.03%	-----

Inception V3



https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Szegedy_Rethinking_the_Inception_CVPR_2016_paper.pdf

<https://paperswithcode.com/method/inception-v3>

05

Conclusiones

¿Qué se logró con el proyecto?

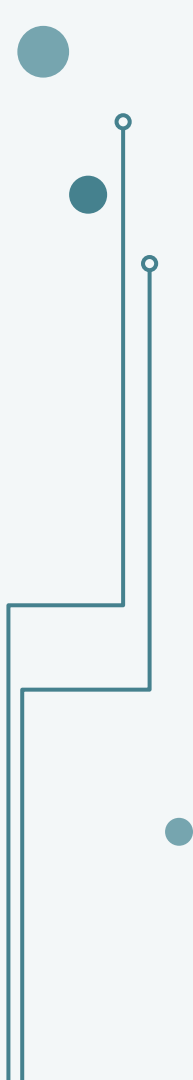




Conclusiones

Finalmente...!

Se logró clasificar los defectos de las placas de circuitos impresos haciendo uso de la mayoría de arquitecturas expuestas en clase, sin embargo, la inception V3 destaca sobre todas las arquitecturas al tener los mejores resultados.



¡Muchas Gracias!

