



### Classifying Interior Design Styles Using CNN

PROJECT PRO5 BY:
NADIA & MANAL



start



### CONTENTS

Introdution

Data & Process

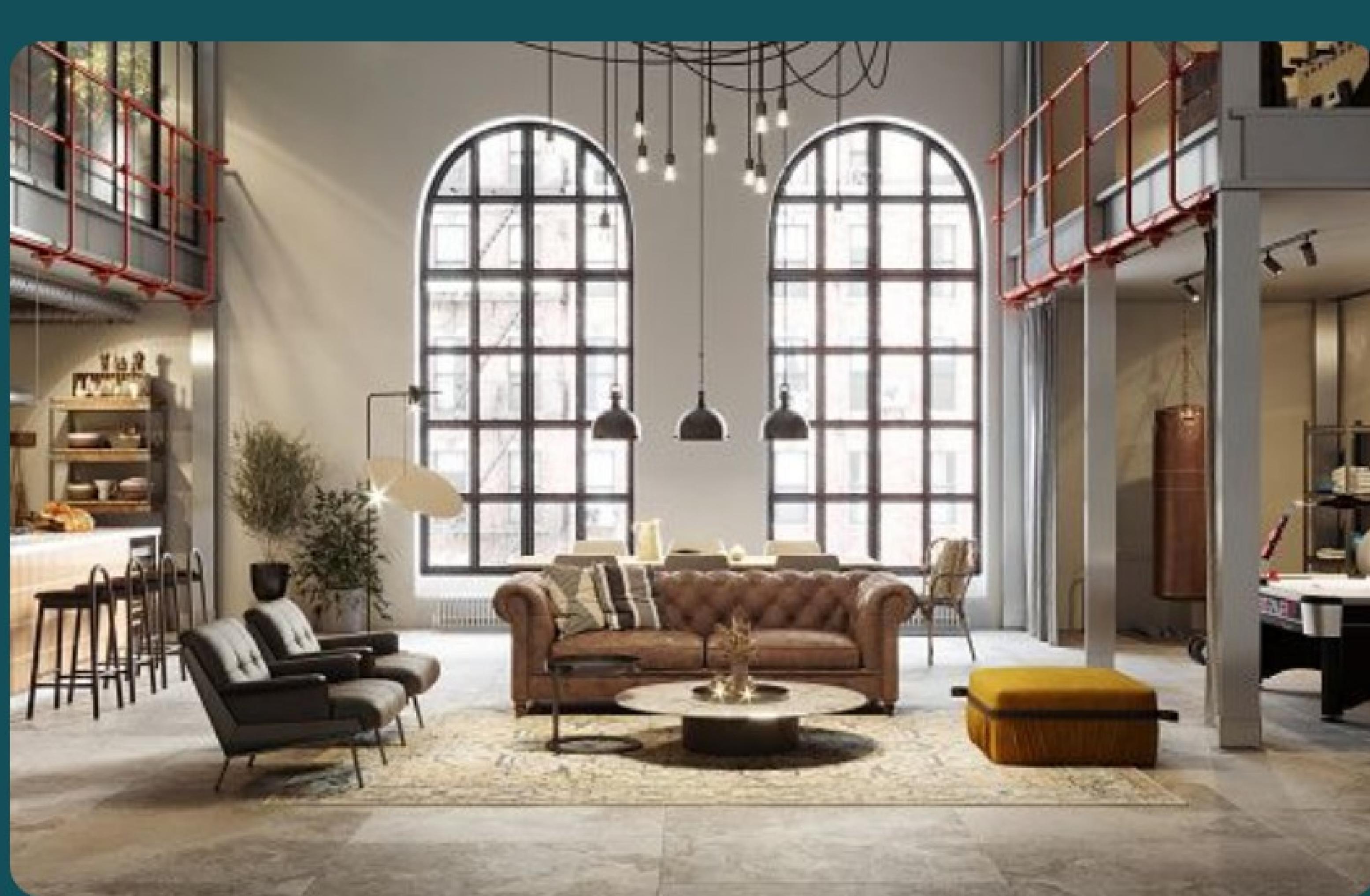
Model & Result

Evaluation & Conclusion

### Introduction

Interior design involves a high amount of guessing. Although a room's style can be predefined and categorized, these are typically hard to classify it by non experts. Thus, our goal is to classify some interior designs for different rooms based on their style; we focus on the two primary styles: modern, traditional (old).

To achieve this goal, we utilized deep neural networks.















Keras

## Data

Data source: kaggle [https://www.kaggle.com/robinreni/house-rooms-image-dataset].

Bedroom : 1248
Dinning : 1158
Bathroom : 606
Livingroom : 1273
Kitchen : 965

### Process



#### Preparing data

" label dataset "

#### Preprocessing

- Train-test split
- image preprocessing:
  - 1- Augmetation

#### Train the model

transfer learning:

- 1- Feature extraction
  - 2- fine Tunning

#### Result

chose the best performance model

## Feature Extraction & Result

Model: "model"				
Layer (type)	Output	Shape	Param #	Connected to
input_1 (InputLayer)	[(None	, 224, 224, 3)	0	
rescaling (Rescaling)	(None,	224, 224, 3)	0	input_1[0][0]
normalization (Normalization)	(None,	224, 224, 3)	7	rescaling[0][0]
stem_conv_pad (ZeroPadding2D)	(None,	225, 225, 3)	0	normalization[0][0]
stem_conv (Conv2D)	(None,	112, 112, 32)	864	stem_conv_pad[0][0]
avg_pool (GlobalAveragePooling2	(None,	1280)	0	top_activation[0][0]
batch_normalization (BatchNorma	(None,	1280)	5120	avg_pool[0][0]
dropout (Dropout) ][0]	(None,	1280)	0	batch_normalization[0
dense (Dense)	(None,	2)	2562	dropout [0] [0]
Total params: 4,057,253 Trainable params: 4,012,670 Non-trainable params: 44,583				

- + Baseline Model (EfficientNetB0):
  - Imagenet weights
  - epochs = 25
  - learning rate = 0.001
  - Batch size = 32
  - Loss = 'binary\_crossentropy'
  - Optimizer = ADAM

	precision	recall	f1-score	support	
M 0	0.77 0.89	0.83 0.84	0.80	129 204	
accuracy macro avg weighted avg	0.83 0.84	0.84 0.84	0.84 0.83 0.84	333 333 333	

loss: 0.3902 auc: 0.9226



- Imagenet weights
- -epochs = 50
- learning rate = 0.000005
- Batch size = 32
- Loss = 'binary\_crossentropy'
- Optimizer = ADAM

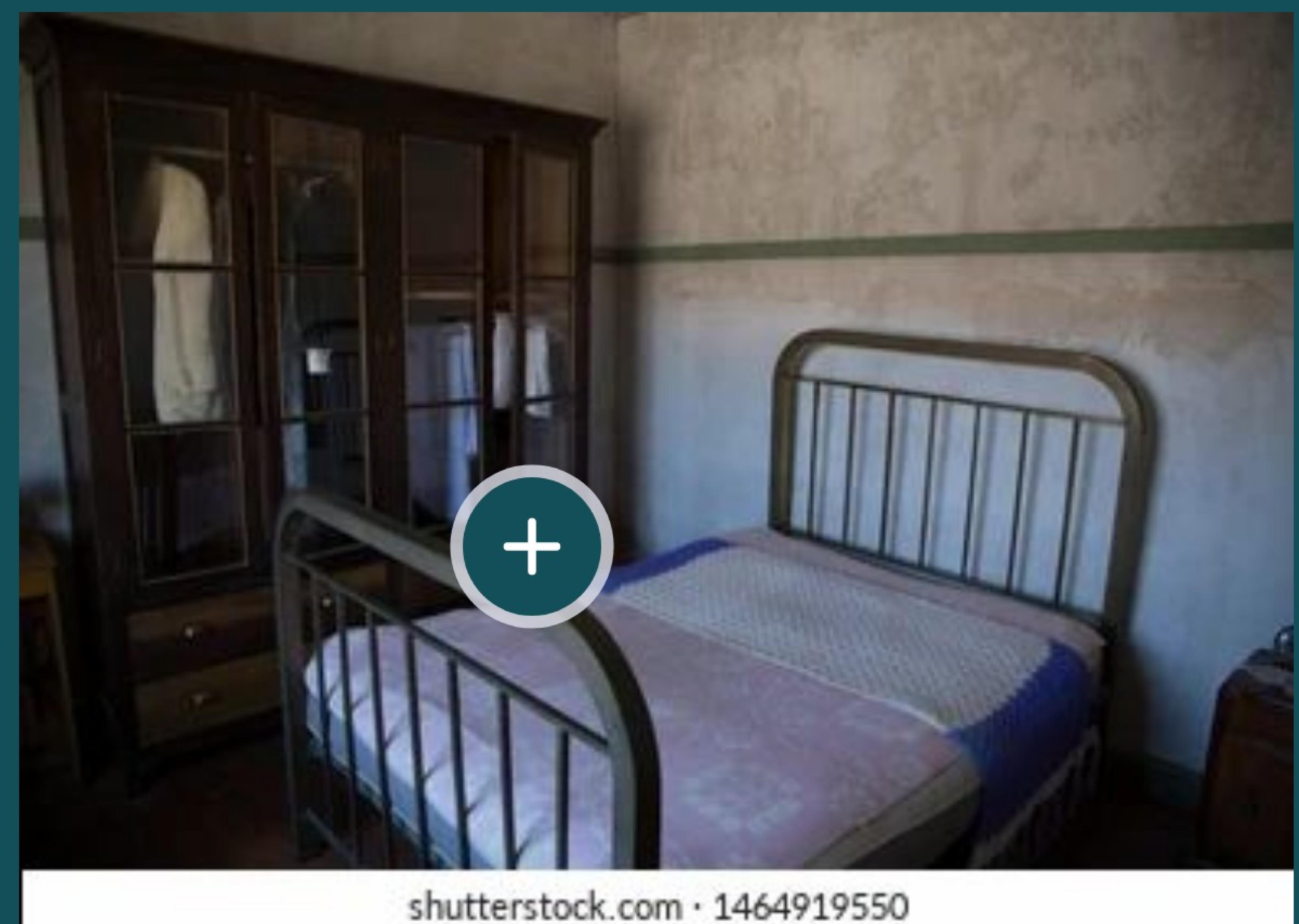
	precision	recall	f1-score	support
M 0	0.99 0.97	0.95 1.00	0.97 0.98	129 204
accuracy macro avg weighted avg	0.98 0.98	0.97 0.98	0.98 0.98 0.98	333 333 333

loss: 0.0906

auc: 0.9927

# Evaluating the Model on New Images





# Evaluating the Model on New Images





+ This model has 50 cycles, if we change the epoch to a big number we can get a more precise results

