Classification & Localization of Inpainted Regions

Deep Learning | MSc Artificial Intelligence

Andreas Sideras

NCSR Demokritos & University of Piraeus

June 2023

Outline

Table of Contents

Problem Definition

- The goal is to train a model that detects an artificial region in an image.
- An artificial region refers to a part of the image that has been generated by AI.
- Each image may or may not have been edited, which leads us to approach the problem as an object detection task, consisting of Classification and Localization subtasks.
- Most publicly available datasets primarily consist of fully generated images. Therefore, we have generated our own dataset, which includes images that may or may not contain an artificial region.

Table of Contents

Image Inpainting

- Image Inpainting refers to the process of filling in missing or corrupted parts of an image based on the surrounding context.
- GANs can be trained to generate realistic and visually coherent content to fill in missing or damaged areas of an image.
- We perform Image Inpating in the Places365 (a small subset of) dataset in order to create data for our task.
- "Free-Form Image Inpainting with Gated Convolution" proposes a novel model architecture, that demonstrates excellent performance on this task.

Dataset generation

- The model requires two images as inputs. An image with the region to be inpainted covered by a mask and an image that includes just this mask. Then, it outputs the inpainted image.
- We created a python script that creates such boxes. Their position and size are distributed uniformly.
- We mixed inpainted and non-inpainted images in a dataset that sums up to 36500 images.
- Each image corresponds to a 5-dimensional ground truth vector (inpainted? , xmin, ymin, xmax,ymax).

Dataset generation

(a) Original

(b) Mask

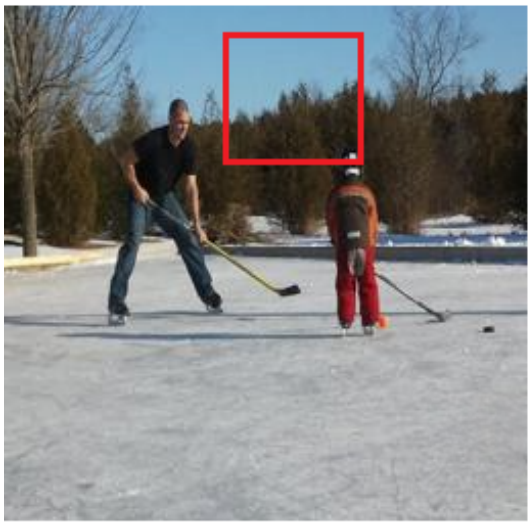
(c) Masked

(d) Inpainted image

Dataset generation




Dataset generation



Dataset generation



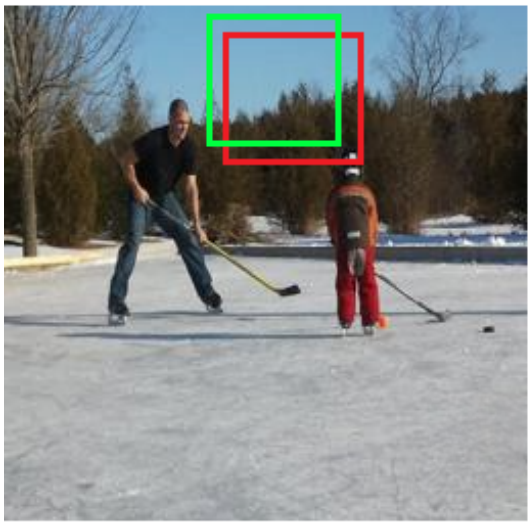
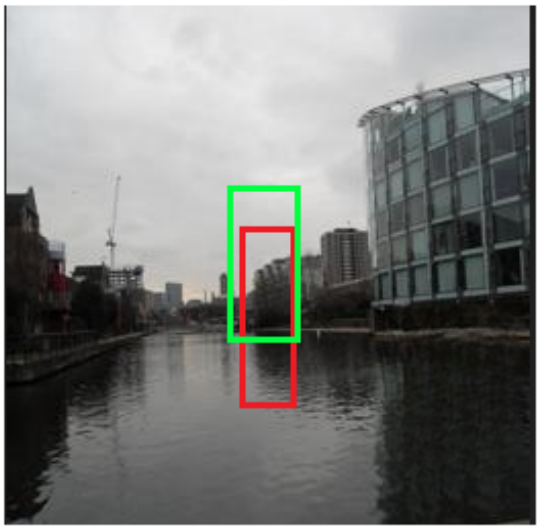



Table of Contents

Classification & Localization

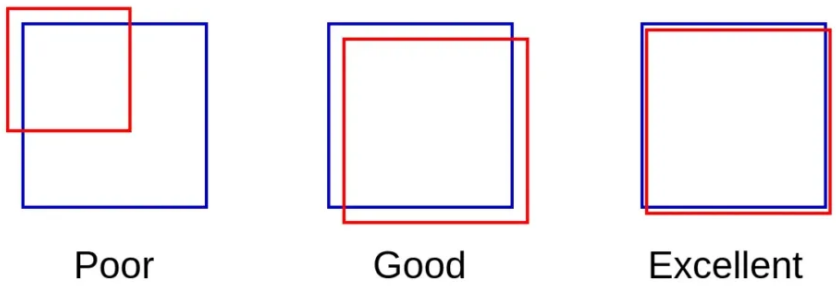
- We used pretrained (on ImageNet) state of the art CNN models.
- We finetuned them after changing their last layer to match our representation, so they consist of one sigmoid and 4 linear units. The fist one will address the binary classification task and the rest will perform the bounding box regression.
- Both tasks have to be solved simultaneously, so we need one common loss function:

$$\text{Total Loss} = \alpha \cdot \text{Classification Loss} + \beta \cdot \text{Localization Loss}$$

How to measure performance?

- For the binary classification part we have some pretty standard options like Precision, Recall, f1 etc.
- Intersection Over Union (IoU) $\in [0, 1]$ is a *scale invariant* metric for the assessment of the bounding box regression.

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$



Scale Invariant property

Pair 1:

$$\text{Ground Truth Bbox:} \quad (x_{\text{min}} = 10, y_{\text{min}} = 10, x_{\text{max}} = 80, y_{\text{max}} = 80)$$
$$\text{Prediction Bbox:} \quad (x_{\text{min}} = 15, y_{\text{min}} = 15, x_{\text{max}} = 85, y_{\text{max}} = 85)$$

Pair 2:

$$\text{Ground Truth Bbox:} \quad (x_{\text{min}} = 10, y_{\text{min}} = 10, x_{\text{max}} = 180, y_{\text{max}} = 180)$$
$$\text{Prediction Bbox:} \quad (x_{\text{min}} = 15, y_{\text{min}} = 15, x_{\text{max}} = 200, y_{\text{max}} = 200)$$

IoU(Pair 1) $\approx$ IoU(Pair 2) ≈ 0.75

MSE loss(Pair 1): $= \frac{(10-15)^2+(10-15)^2+(80-85)^2+(80-85)^2}{4} = 25$

MSE loss(Pair 2): $= \frac{(10-15)^2+(10-15)^2+(200-185)^2+(200-185)^2}{4} = 225$

Loss Functions

- Standard Binary Cross Entropy Loss for classification.
- IoU cannot be used as localization loss directly, because it suffers from zero gradient computations.
- MSE, mean L2/L1 distance and Smooth-L1 are common choices.
- Generalized-IoU is a derivation of IoU that has non-zero gradient everywhere.

$$\text{smooth L1}_\beta(\hat{y}, y) = \begin{cases} 0.5 \cdot (\hat{y} - y)^2 & \text{if } |\hat{y} - y| < \beta \\ |\hat{y} - y| - 0.5 \cdot \beta^2 & \text{otherwise} \end{cases}$$

$$\text{GIoU} = \text{IoU} - \frac{|\text{C}/(A \bigcup B)|}{|C|}$$

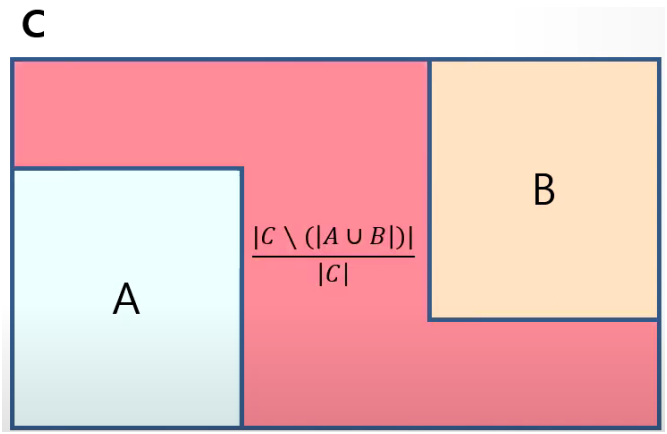


Table of Contents

Experiments

Table: Models' Performance on Test set (6500 images)

Model	Precision	Recall	F1	Mean IoU
AlexNet	0.878	0.852	0.8657	0.531
VGG11	0.882	0.897	0.8955	0.545
ResNet18	0.902	0.913	0.908	0.688
ResNet50	**0.928**	**0.99**	**0.958**	**0.713**

Best model

Table: ResNet50 best model hyperparameters

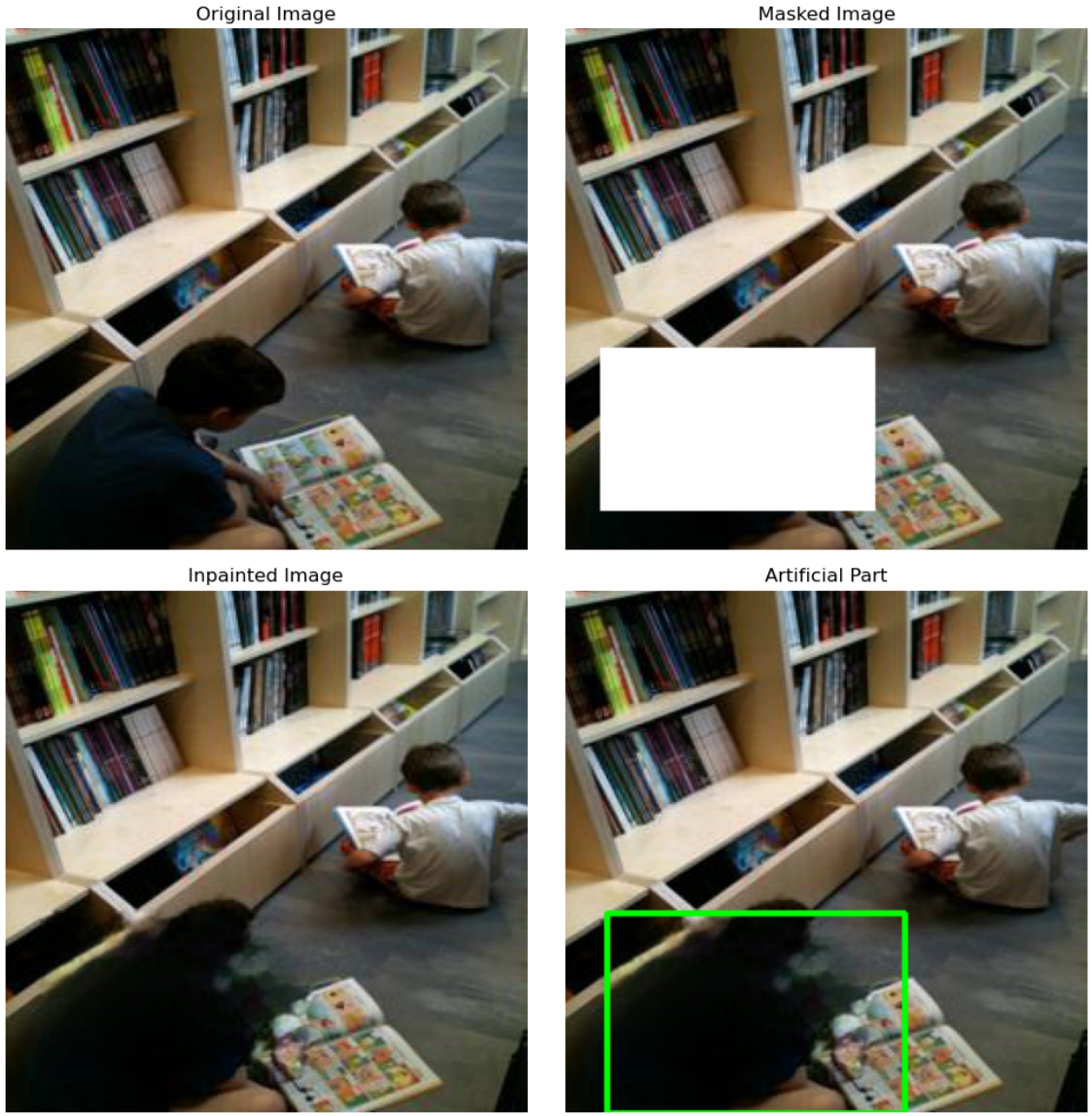
Hyperparameter	**Value**
NUM_EPOCHS	7
BATCH_SIZE	512
LEARNING_RATE	0.001
ALPHA	100
BETA	0.8

Table of Contents

Performance analysis

- The Classification task were very easy for the models, localization proved harder.
- All the models have exhibited high accuracy in cases where the GAN network generated blurry outputs (and discontinuities). Even in instances of minimal blurriness, imperceptible to the human eye, a neural network can effortlessly detect it.
- By artificially generating regions with a fixed rectangular shape, we introduce a strong bias that the models are capable of capturing. Within these rectangles, the pixels generated by the GAN, may exhibit similar value distributions.
- These steep horizontal and vertical transitions may be challenging for the human eye to distinguish, but neural networks find them relatively easy to detect.
- Idea: Transform the problem into Classification & Segmentation.

An easy result



A good result

Thank you!