UNO Card Object Detection using Faster R-CNN

Alessio De Luca 919790
Davide Vettore 868855

1. Objective:

The goal of this project was to train a Fast R-CNN model capable of performing object detection on UNO cards, in particular we wanted to correctly identify all the 14 numbers and symbols present on the cards. In order to evaluate the model's performance, we used the *Intersection over Union* score (IoU). This metric measures how well a predicted object bounding box aligns with the actual object bounding box.

2. Introduction:

The dataset used for this project is composed of 8,992 images of UNO cards and 26,976 labeled examples on various textured backgrounds.

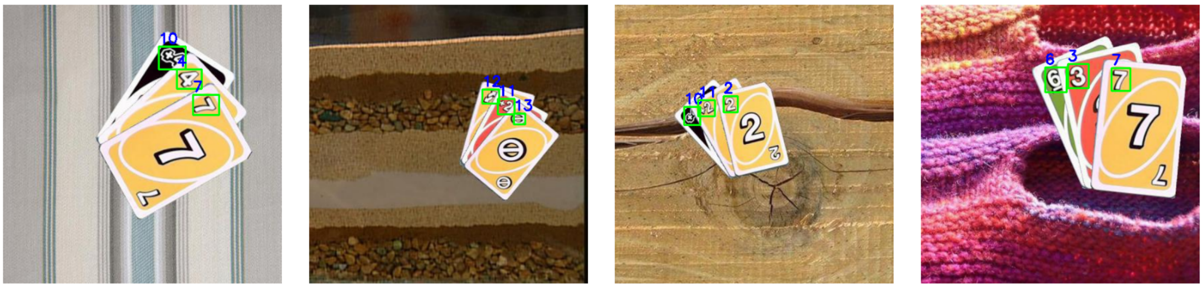

Figure 1: Illustration of 4 images used for training with the associated bounding boxes.

The Neural Network used to perform the object recognition task is **ResNet-50**, a Faster R-CNN model with a ResNet-50-FPN backbone based on a 2016 paper[1] from Shaoqing Ren *et al.* While R-CNN are very effective in object recognition, they suffer from slow inference speed due to their sequential processing of region proposals. Faster R-CNN insert a *Region Proposal Network* (RPN) after the last convolutional layer. RPN are trained to produce region proposals directly, without the need for external region proposals, a criticality in classic R-CNN. In particular, ResNet-50 is a deep CNN with 50 levels of depth (48 *convolutional layers*, one *max pool layer*, and one *average pool layer*). Pre-trained on a dataset exceeding 1 million images from ImageNet, this model has been primed to classify objects across 1000 classes with notable accuracy.

The performance of the model was then valuated using the *Intersection over Union* (IoU) metric: it is used to evaluate the accuracy of annotation, segmentation, and object detection algorithms. It quantifies the overlap between the predicted bounding box or segmented region and the ground truth bounding box or annotated region from a dataset. IoU provides a measure of how well a predicted object aligns with the actual object annotation, enabling the assessment of model accuracy and the fine-tuning of algorithms for improved results.

[1] Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.

3. Procedure:

After loading the dataset, all the possible features of UNO cards were assigned a label: the background was labeled as ‘background’, numbers from 0 to 9 were labeled with their same number and special features were labeled from 10 to 14, respectively “+4” as ‘10’, “+2” as ‘11’, “Reverse” as ‘12’, “Skip” as ‘13’ and “Wild” as ‘14’.

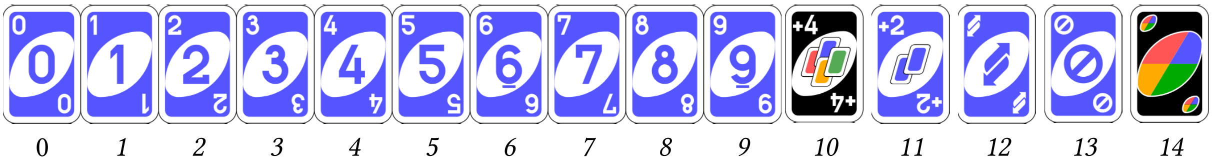

Figure 2: Illustration of all UNO cards with their associated labels.

Preprocessing techniques such as resizing, normalization and augmentation were computed: all images were resized to 416x416 format and all the pixels values were divided by 255. Images were also flipped, rotated and blurred and then converted from BGR to RGB color format. All bounding boxes coordinates provided in the XML files were adjusted to match the new resized format of the images. The dataset was then split in 3 subsets: 6295 images (70% of the total) for training, 1798 (20%) for validation and 899 (10%) for the final testing.

Once the dataset preparation was completed, the ResNet-50 model was defined, initialized with pre-trained weights derived from extensive training on the ImageNet dataset. These pre-trained weights act as a foundation, offering the model valuable visual representations learned from various images. The model was then ready to undergo training in which the weights were fine-tuned using Adam optimizer with a learning rate of 0.0001.

During each training epoch, the model iterates through the training dataset in batches of 8 images, adjusting its parameters to minimize the loss function. The model’s performance was evaluated on the validation set by calculating the validation loss.

Once all the training epochs were completed, the model’s final performance was assessed on the test set calculating the average IoU. Each IoU was obtained dividing the intersected area between predicted and ground truth bounding boxes over the union of the same two areas, resulting in a value between 0 (worst case) and 1 (perfect case).

The function defined to compute the average IoU of each test image accepts a boolean parameter which determines if the boxes *not* caught by the model should contribute to the average IoU as a zero. This highly penalizes the average IoU in the images where not all the boxes are caught, the results shown includes this penalization.

4. Results:

After only one training epoch the value of the validation loss was already very low, equal to 0.1645. This shows how well the fine-tuned ResNet-50 model is in performing object detection. The results on the test set confirm this observation, with the average IoU score for all images being 87.7%

Our model extracts multiple boxes for each image, by setting a detection threshold we decided to keep only boxes with confidence score higher than 80%.

In order to compute how fast the model performs we compute the average frames processed by the model in one second (*frame per second*). On the 113 test subset we achieved on average 9.43 fps.

In Figure 3 and Figure 4 the four images with the highest and lowest average IoU respectevly are shown.

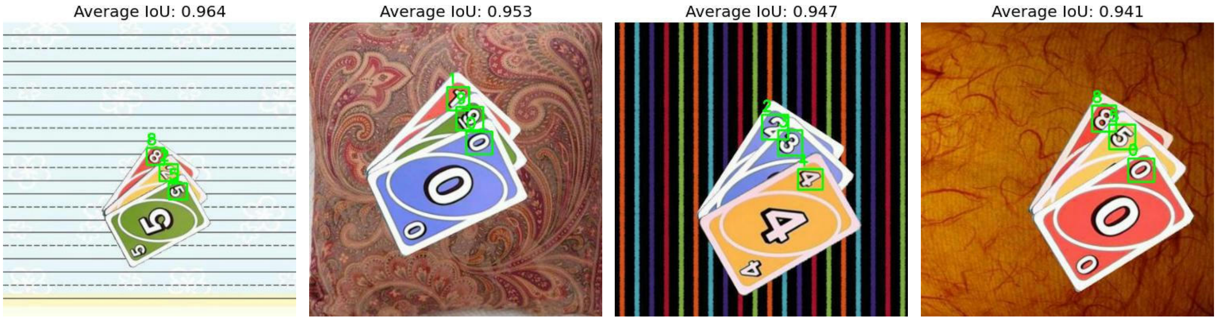

Figure 3: Illustration of the images with the 4 highest average IoU values

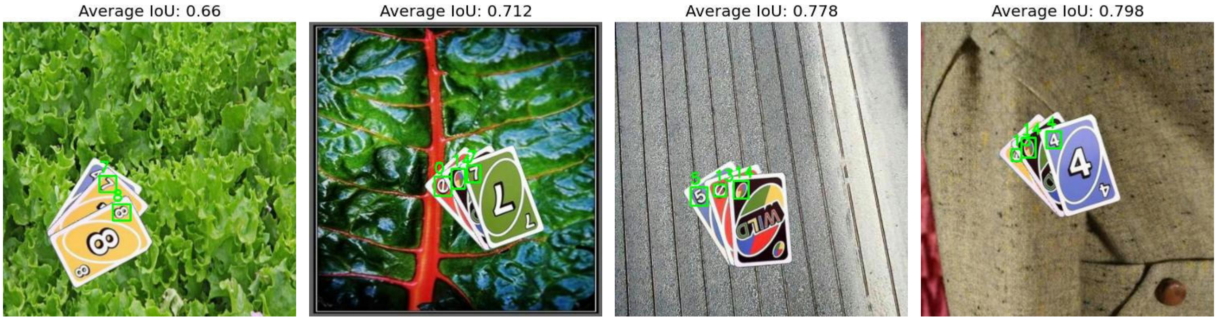

Figure 4: Illustration of the images with the 4 lowest average IoU values

5. Conclusions:

In conclusion, our project has successfully demonstrated the effectiveness of training a sophisticated ResNet-50 model for UNO cards object recognition.

Examining the results reveals that the model missed only one bounding box across the entire test set. As previously mentioned, the failure to detect all boxes significantly impacts the average Intersection over Union metric, as the IoU value for a missed bounding box is zero. Consequently, it was anticipated that images with undetected boxes would yield the lowest average IoU. In cases where all the boxes are detected, the predominant factor affecting the IoU seems to be the extent to which labels are obscured by foreground cards. It's worth noting that in images with the highest average IoU, all labels are fully visible.

Furthermore, our assessment of processing speed, averaging 9.43 frames per second, highlights the model's potential for real-time applications.