Soccer Robot Perception
Lab Vision Systems

Kumar Abishek, Rajan Heena

Universität Bonn
s6abkuma@uni-bonn.de, Matrikelnummer: 3205919
s6heraja@uni-bonn.de, Matrikelnummer: 3205531

Abstract. Object detection and semantic segmentation are two important aspects in the field of computer vision. There has been a lot of research to solve both tasks using Convolutional Neural Networks. These approaches give good results but are computationally demanding. Also, even though both the tasks are correlated, they are typically solved separately with different techniques. This paper describes a unified framework for object detection and semantic segmentation for soccer-related objects and humanoid robots under different lighting conditions.

1 Introduction

In recent times, research in the field of computer vision has proceeded at a break neck pace. Several works have been done on image recognition, object detection, image generation and semantic segmentation. Both object detection and semantic segmentation are important aspects of computer vision due to several practical use cases. In applications such as self-driving cars, face recognition and medical image processing, the image content is analyzed by performing following basic tasks like image classification, object detection, segmentation and so on. It helps to gain knowledge of a given scene image. For example, an autonomous car needs to delimitate the roadsides with a high precision to move by itself and semantic segmentation helps by discovering the driveable areas on an image.

This project aims to detect the soccer-related objects such as soccer ball, goalposts, robots, field boundaries and line segments in extremely low and very bright lighting conditions. We use a unified convolutional network to achieve this.

1.1 Object detection

It is the process of finding the location of object instances of one or more object classes present in an image regardless of view with respect to the camera, partial occlusions, and illumination conditions. It analyzes complex images that contain objects at different distances and locates and classifies unknown objects, as well as determine their boundaries.

1.2 Semantic segmentation

Semantic segmentation is one of the techniques to understand an image at a pixel level. It is the task of classifying each pixel in an image into a class. Specifically, it attempts to partition the image into semantically meaningful parts and to classify each part into one of the pre-defined classes.

2 Related Work

2.1 SegNet

It is a deep encoder-decoder architecture for multi-class pixelwise segmentation [2]. It consists of a sequence of encoder layers and respective decoder layers. The encoder is made of convolutional layers followed by ReLU, max-pooling and sub-sampling. The decoder up-samples the image using the max-pooling indices. This helps in retaining high-frequency details in the segmented images. SegNet was developed for outdoor and indoor scene understanding.

2.2 U-Net

It was first designed especially for medical image segmentation. This network is based on the fully convolutional network and consists of three sections: The contraction, The bottleneck, and the expansion section. In U-Net, feature maps used for contraction are used to expand a vector to a segmented image [4]. This preserves the structural integrity of the image which would reduce distortion enormously. The encoder consists of the typical convolutional network made of multiple convolutions followed by ReLU and max pooling operations. It has the addition of skip connections which allow a more refined and precise output.

3 Our Approach

Object detection and segmentation are important tasks for visual perception in events like RoboCup. The main objective is to recognize soccer-related objects including a soccer ball, robots, line segments, field boundaries and goalposts. This goal is achieved by using a unified deep convolutional neural network to perform object detection and pixel-wise classification with one forward pass [3]. This network is also robust against low and bright lighting conditions.

3.1 Network Architecture

The unified convolutional network has an encoder-decoder architecture similar to pixel-wise segmentation models like SegNet [2] and U-Net [4]. This architecture is illustrated in Fig.1. A pre-trained ResNet-18 model is used as an encoder and it is a stack of convolutional, batch normalization and ReLU layers. The

Global Average Pooling layer and final fully connected layer are removed to retain higher resolution feature maps at the deepest encoder output. The decoder upsamples its input feature map(s) using transpose convolutional layers. It has four convolutional transpose layers with ReLU and Batch Normalization. It is also shorter than the encoder part to minimize the number of parameters in the network and achieve real-time computation. The number of parameters has been reduced at the cost of losing fine-grained spatial information. This trade-off is balanced by using sub-pixel post-processing where contours are detected from the outputs of the network and centers are calculated, comparing them with the original centers. Inspired by the U-Net model, three lateral connections are made from the encoder part to the decoder part to directly provide high-resolution details. The convolutional deep learning architectures are spatially invariant, and therefore cannot recognize location-dependent features. To tackle this issue, we used location-dependent convolutional layer in the last layer of the model to incorporate location-dependent features [1]. The network consists of two output heads; one for object detection and the other for pixel-wise segmentation and both the heads share a common learnable bias to limit the number of parameters used. The model takes 3-channel(RGB) image as input and outputs two images corresponding to both detection and segmentation tasks.

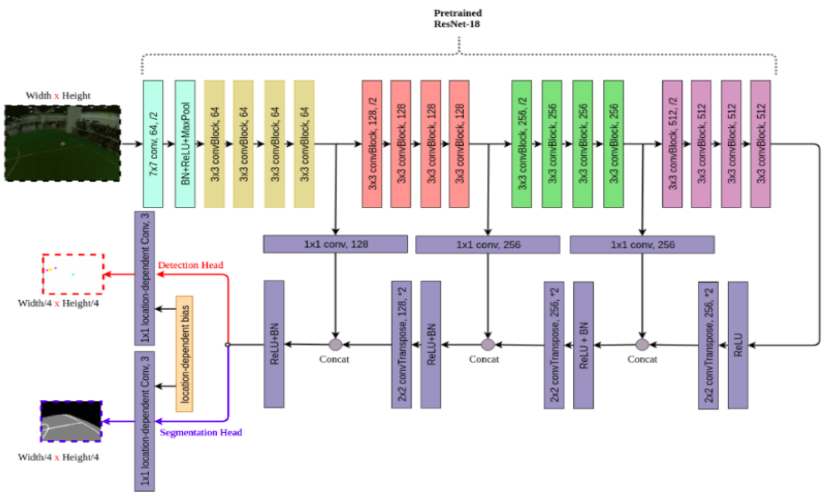

Fig. 1. Network Architecture

3.2 Pre-Processing

In the pre-processing phase, we used excessive data augmentation by applying deformations to the available training images. The dataset is small and these de-

formations significantly improve the diversity of data available. The images are first rescaled to picture size of 640x480 pixels (width * height). Then the following modifications are applied to the images during training:random horizontal flips, random vertical flips and color jitter. This makes the network reasonably robust to position, brightness and color shifts. The normalization of the dataset is performed with the same values used in the Resnet18 pre-trained network. All these transformations are applied to the images and also on the bounding box details wherever applicable. In the case of object detection task, we generated three heat-maps corresponding to three output channels. The target is constructed by Gaussian blobs around the ball center and bottom-middle points of the goalposts and robots. The heat maps are 2D-maps of feature locations and are of size 160*120 i.e. width/4 * height/4 pixels of the input pictures used. In case of segmentation, apart from background, lines and field, the label also contains ball details. We merged ball details with field information in one class. Examples of input images and corresponding ground truth labels for both the tasks are shown in Fig.2.

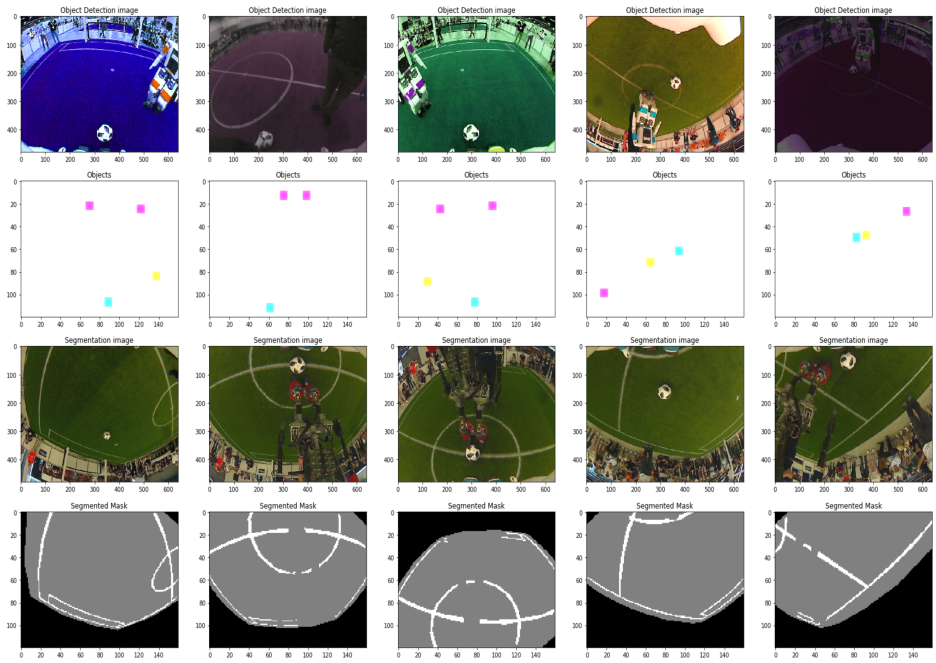

Fig. 2. Top row: input images for detection. Second row: their corresponding heat-maps with balls (cyan), goalposts (magenta), and robots (yellow). Third row: input images for segmentation. Bottom row: their corresponding labels.

3.3 Loss Function

We used two different loss functions for both the tasks. In the object detection head, mean squared error(MSE) is used similar to SweatyNet [5] and in the classification head, we used cross-entropy loss function. We tried using Total Variation loss to the outputs which force the result to have smaller total variation and encourages smoothing. However, the improvement was not prominent.

4 Training

The training is performed on two datasets consisting of 3598 images for object detection and 1108 images for segmentation task. We randomly chose 80 percent of the train datasets to be actual train set and the remaining 20 percent to be validation set. The model is trained for detection and segmentation tasks in alternate batches with a batch size of 8 images. Due to imbalanced datasets, we introduced a hyper-parameter to penalize segmentation loss. We trained the model for 106 epochs using the ADAM optimizer and with learning rate of 0.001. We also tried using cyclic learning rate as proposed by Smith et. al. [6] but the results did not improve significantly. The training was done on a NVIDIA 1050Ti, Intel i7-7700HQ machine and it took around 26 hours. The loss curves of training and validation phases are shown in Fig.3.

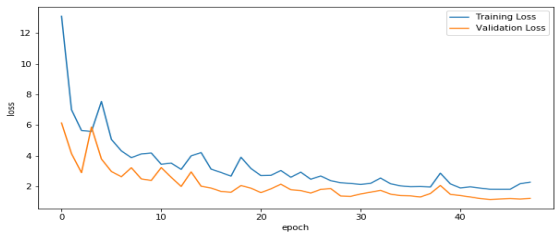

Fig. 3. Learning curves

5 Results

The evaluation is done on two test sets containing 5255 images for detection and 84 images for segmentation tasks. For object detection, a test dataset is prepared containing images along with expected center of the balls, goalposts and robots. The detection head outputs a 3-channel heat-map similar to the ground truth with a Gaussian blob at the center of the identified object. A post-processing approach is followed to identify these blobs in the output using OpenCV contour detection and the center of the contour is then compared to the center of the

bounding box of the input labels. If the euclidean distance between the true center and output center is up to 8 pixels, the output blob is taken as correct prediction. The segmentation head also gives 3-channel image where the lines are depicted in white, fields in gray and background is in black color. Fig 4. shows outputs from both heads.

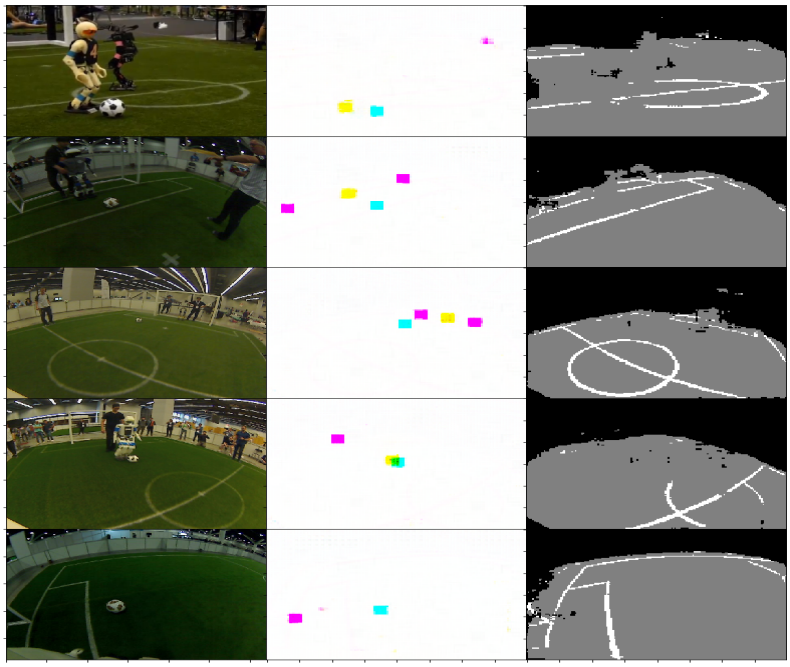

Fig. 4. Detection and segmentation results. Left: input test images. Center: output of detection head with balls (cyan), goalposts (magenta), and robots (yellow). Right: the output of the segmentation head with lines (white), field (gray), and background (black).

5.1 Metric Calculation

The performance of the network is evaluated by calculating the Recall, Precision, F1, Accuracy and False detection rate for soccer-related objects and IoU for segmentation. IoU is the area of overlap between the predicted segmentation and the ground truth divided by the area of union between the predicted segmentation and the ground truth.

$$Recall = \frac{TP}{TP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$FDR = \frac{FP}{TP + FP} \tag{3}$$

$$F1 = 2 * \frac{precision * recall}{precision + recall} \tag{4}$$

$$IoU = \frac{Area\,of\,Overlap}{Area\,of\,Union} \tag{5}$$

True positive: The contour detected by the network is considered to be true positive if the euclidean distance between the center of the contour and center of the bounding box is under 8 pixels.

False negative: When there is a bounding box detail in the true value but it is not detected by the network or detected center of the contour in the output is at a distance more than 8 pixels.

False positive: When there is no boundary box detail in the true value but is detected by the network.

Table 1. Results of object detection.

Type	**F1**	**Accuracy**	**Recall**	**Precision**	**FDR**
Ball	0.86	0.76	0.90	0.83	0.17
Goalpost	0.44	0.29	0.46	0.43	0.57
Robot	0.23	0.13	0.23	0.24	0.76
Total	0.51	0.39	0.53	0.50	0.50

Table 2. Results of semantic segmentation.

Type	**Accuracy**	**IoU**
Background	0.97	0.94
Field	0.95	0.90
Lines	0.98	0.77
Total	0.96	0.87

6 Conclusion and Future Work

We can conclude that the model is able to perform both tasks in one forward pass. Although we are getting good result for segmentation task, there is a room

for improvement for object detection task. We think tuning circular learning rate parameters, balancing the object detection classes and extensive training can improve the result significantly.

References

[1] Niloofar Azizi et al. "Location dependency in video prediction". In: *International Conference on Artificial Neural Networks*. Springer. 2018, pp. 630–638.

[2] Vijay Badrinarayanan et al. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation". In: *IEEE transactions on pattern analysis and machine intelligence* 39.12 (2017), pp. 2481–2495.

[3] Diego Rodriguez et al. "RoboCup 2019 AdultSize Winner NimbRo: Deep Learning Perception, In-Walk Kick, Push Recovery, and Team Play Capabilities". In: *Robot World Cup*. Springer. 2019, pp. 631–645.

[4] Olaf Ronneberger et al. "U-net: Convolutional networks for biomedical image segmentation". In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[5] Fabian Schnekenburger et al. "Detection and localization of features on a soccer field with feedforward fully convolutional neural networks (fcnn) for the Adultsize humanoid robot Sweaty". In: *Proceedings of the 12th Workshop on Humanoid Soccer Robots, IEEE-RAS International Conference on Humanoid Robots, Birmingham*. sn. 2017.

[6] Leslie N Smith. "Cyclical learning rates for training neural networks". In: *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE. 2017, pp. 464–472.