Abstract

In this work, we explore an end-to-end deep learning approach to spatially and temporally localizing actor-object relationships in videos via an encoder-decoder convolutional neural network; the end-goal of such a network is to output spatiotemporally localized tubes with a triplet label (object, verb, subject), but the implementation of the latter is only hypothesized in this paper. We also touch on some of the VidOR [1] dataset and how to address them.

1. Introduction

The goal of this project is to perform joint pixel-wise identification of objects, the actions they perform, and the relationships among them, providing temporal and spatial localization for each label, and establishing subject & object roles.

The task builds on action and object segmentation that many state-of-the-art approaches, such as R-CNNs [2], tended to achieve via Region Proposal Networks (RPNs)[2], which break the end-to-end flow of the network and so are generally not preferred.

As opposed to RPN models, our end-to-end encoder-decoder model outputs per-frame pixel-wise class-segmentation of the input to retrieve spatiotemporal localizations of the actions & objects and performs instance separation in post-processing, which retains the benefits of end-to-end training.

2. Related Works

This project builds on two common tasks in computer vision: spatiotemporal action & object localization and actor/object relation identification.

1.1. Spatiotemporal Detection and Localization

In the past few years, there has been a lot of work done in spatiotemporal detection and localizations of actions.

Most notably, the discovery of R-CNNs [2] has introduced an effective way of extracting regions that are most likely to contain foreground using a Region Proposal Network (RPN), therefore reducing the overall space for object classification in images.

![T-CNN Network](image)

Figure 1 – T-CNN Network [3]

Later, T-CNNs [3] (Figure 1) were introduced as the generalization of R-CNNs, which uses a Tube Proposal Network (TPN) as the generalization of RPN in the temporal dimension. Though these approaches work well, they aren’t truly end-to-end.

Our project aims to eliminate the RPN and its variations in order to make the training truly end-to-end.

1.2. Actor/Object Relation

Actor/object relation is a lot less researched topic in computer vision, but there were quite a few notable approaches in the past few years as well.
Facebook AI Research (FAIR) released a paper on a human-centric variation of the problem, which aimed to identify the relation between humans and objects [4]. Using a variation of the R-CNN -- Faster R-CNN [5] as its backbone, the network outputs three branches: object detection branch, human-centric branch, and interaction branch (Figure 2). The first branch simply detects the humans and objects in the picture, the second branch generates a gaussian region of where the target of a person’s action may be, based on said person’s posture and position, and the third branch combines the first two to output the (object, action, subject) triplet [4].

Though FAIR’s network works quite well, its scope is limited to images with humans, while our project extends the scope to interactions among animals and objects in videos. Finally, FAIR’s network utilizes an RPN variation, which we aim to avoid.

3. Approach

Our approach is to output per-frame actor, object, and action segmentations which provide both temporal classification and localization for the clip along with binary centroids, which will allow us to perform instance separation on the segmentation.

1.3. Structure

The model’s structure consists of two parts: an encoder and a decoder. We use I3D [6] as our encoder, as it provides state-of-the-art video feature extraction. After the features are extracted, our model is split into three decoder branches: actor & object segmentation, action segmentation, and binary centroid segmentation (Figure 3).

The decoder uses deconvolutions (transpose convolutions) for up-sampling and Atrous convolutions [7] for fine-feature extraction, as they allow to cover larger convolution areas without increasing the number of learnable parameters.

Due to the class imbalance discussed later, each branch was trained separately in object, action, centroid order.

1.4. Dimensionality

To gain a better grasp of the flow, the input of the network is (16, 224, 224, 3), where 16, is the number of frames, (224, 224) is the dimension of the frames, and 3 are the RGB channels. The encoder proceeds to downsample the dimensions, while extracting features, resulting in a (16, 14, 14, 256) receptive field by the end of the extraction. The extracted features are then split into three branches and up-sampled to (16, 56, 56, d), at which the losses are calculated. The last dimension d is the respective number of classes there are for each branch, while the centroids segmentation is a binary label, which simply predicts the presence of an object center.

1.5. Post-Processing

In order to measure our performance, we form per-class blobs out of the connected segmentation, filter blobs with areas smaller than a threshold, and take the min-max to get bounding box corners. These bounding boxes are then used to compute mean average precision (m-AP) and mean intersection over union (m-IOU) between the labels and the prediction.

Due to the time constraint, we weren’t able to implement the proposed instance separation using centroids and form actor-action pairs using overlap, but our proposed approach was to separate each action ‘a’ into two different labels ‘a-object’ and ‘a-subject’, which would imply whether the object is a subject or an object of action ‘a’.

After the action segmentation was calculated, the actions would be assigned to actors/items via overlap calculation and combined into subject-object pairs.

4. Dataset

The VidOR dataset, compiled with 80 actors and objects, such as ‘adult’, ‘child’, ‘dog’, ‘toy’, etc., 42 subject-object actions, such as ‘watch’, ‘feed’, ‘drive’, etc., and 8 relations, such as ‘next_to’, ‘behind’, ‘away’, etc.
The dataset uses bounding boxes with object, instance, and action/relation IDs. Though the labels are bounding boxes, during training, we predict the segmentation, which introduces a certain margin of error when calculating accuracy and loss. For the same reason, the area of the segmentation is often overpredicted, but this can be addressed with a higher threshold, with the obvious sacrifice of lower-confidence predictions.

1.6. Challenges

One of the biggest challenges with VidOR is the class imbalance in objects and actions, which can be seen in Figures 4 and 5 respectively. We addressed this issue with class weights, calculated based on the number of frames per class, and class subsampling, though these approaches did not solve the issue entirely. We also opted to remove the “watch” and “lean_on” labels, as these could be applied most of the time there was a living creature and due to their general ambiguity.

5. Results

Though we weren’t able to achieve a mean average precision score for action localization bounding boxes over 0.04 with intersection over union threshold of 0.5, this deviation is likely due to two reasons: class imbalance and inconsistent/noisy labeling in the dataset.

The first reason stems from the fact that, despite subsampling, some of the less common actions have a lot less training diversity, which causes overfitting and, therefore, very low mean-AP scores on the validation set. For example, mean-AP scores of the actions ‘hold’ and ‘play(instrument)’ is close to 0.8, while “release”, “use”, “push”, and “throw”, all much less-common actions, yield a mean-AP score of 0.

The second reason comes from the fact that the bounding boxes in the dataset were auto-generated, so many of the samples are either missing labels or have labels with missing or incorrect classification. An example of a missing label can be seen in Figure 6, where the toddler is holding a baby and talking to it, though the labels only reflect the action of speaking.

Despite these challenges and low quantitative scores, we were able to achieve some impressive qualitative results, which can be seen in figures 7 through 11.

Figure 7 shows the model generalizes well for examples where a full frame of a human is not present, though it does fail to predict the second label of “cleaning”, likely because cleaning is a human-centric action.
Figure 8 show the model correctly predicting speaking, but overpredicting “shake_hand_with”, as it sees an extended arm that is often associated with shaking hands, which likely stems from class imbalance.

Figure 9 is an example of a good prediction on a multilabel sample, though it mixes up “lick” and “bite,” which have very similar visual features and the overall prediction area is quite high.

Figure 10 is an example where the model correctly predicts the action, but the segmentation is weak, likely due to a noisy background, while Figure 11 shows a similar outcome, but with a label that is different from ground truth, though still acceptable.

6. Conclusion

The qualitative results we got are promising, yet it is hard to judge whether the lacking quantitative results are due to dataset inconsistency or a problem with our approach, as there are no baselines on it. It will be interesting to explore this question in the future.

References