CV PROJECT REPORT

PIXELWISE INSTANCE SEGMENTATION WITH A DYNAMICALLY INSTANTIATED NETWORK

- by Anurag Arnab and Philip H.S. Torr

https://github.com/Prathyusha-Akundi/Pixelwise_Instance_Segmentation.git

PRATHYUSHA AKUNDI 2018701014
SOWMYA AITHA 2018702007
TANU SHARMA 2018702012

ABSTRACT

Semantic segmentation and object detection research have recently achieved rapid progress. However, the former task has no notion of different instances of the same object, and the latter operates at a coarse, bounding-box level. The proposal is an Instance Segmentation system that produces a segmentation map where each pixel is assigned an object class and instance identity label. Most approaches adapt object detectors to produce segments instead of boxes. In contrast, the method is based on an initial semantic segmentation module, which feeds into an instance subnetwork. This subnetwork uses the initial category-level segmentation, along with cues from the output of an object detector, within an end-to-end CRF to predict instances. This part of the model is dynamically instantiated to produce a variable number of instances per image. The end-to-end approach requires no post-processing and considers the image holistically, instead of processing independent proposals. Therefore, unlike some related work, a pixel cannot belong to multiple instances.

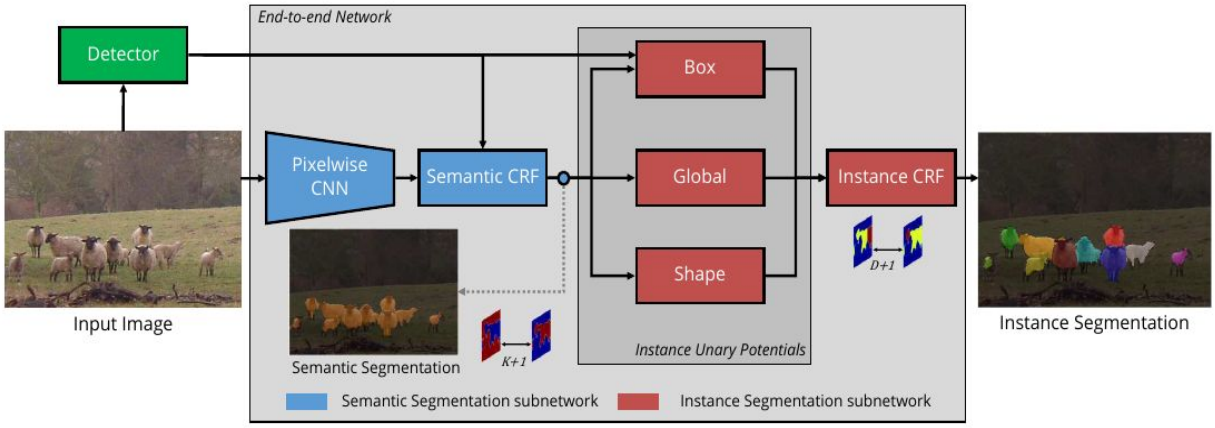
INTRODUCTION

Semantic segmentation and object detection are well-studied scene understanding problems, and have recently witnessed great progress due to deep learning. However, semantic segmentation – which labels every pixel in an image with its object class – has no notion of different instances of an object. Object detection does localise different object instances, but does so at a very coarse, bounding-box level. Instance segmentation localises objects at a pixel level, and can be thought of being at the intersection of these two scene understanding tasks. Unlike the former, it knows about different instances of the same object, and unlike the latter, it operates at a pixel level. Accurate recognition and localisation of objects enables many applications, such as autonomous driving image-editing, and robotics.

The proposed method is inspired by the fact that instance segmentation can be viewed as a more complex form of semantic segmentation, since it is not only required to label the object class of each pixel, but also its instance identity. This model produces a pixel wise segmentation of the image, where each pixel is assigned both a semantic class and an instance label. The end-to-end trained network, which outputs a variable number of instances per input image, begins with an initial semantic segmentation module. The following, dynamic part of the network, then uses information from an object detector and a Conditional Random Field (CRF) model to distinguish different instances. This approach is robust to false-positive detections, as well as poorly localised bounding boxes which do not cover the entire object, in contrast to detection-based methods to instance segmentation. Moreover, as it considers the entire image when making predictions, it attempts to resolve occlusions between different objects and can produce segmentation maps as in without any post-processing.

PROPOSED APPROACH

Network Architecture:



The network contains an initial semantic segmentation module. The semantic segmentation result is used, along with the outputs of an object detector, to compute the unary potentials of a Conditional Random Field (CRF) defined over object instances. Mean field inference in this random field is performed to obtain the Maximum a Posteriori (MAP) estimate, which is the labelling. Although the network consists of two conceptually different parts – a semantic segmentation module, and an instance segmentation network – the entire pipeline is fully differentiable, given object detections, and trained end-to-end.

Semantic Segmentation Subnetwork:

Semantic Segmentation assigns each pixel in an image a semantic class label from a given set, *L*. In our case, this module uses the FCN8s architecture which is based on the VGG ImageNet model. For better segmentation results, we include mean field inference of a CRF as the module's last layer. This CRF contains the densely-connected pairwise potentials and is formulated as a recurrent neural network. Additionally, we also include the Higher Order detection potential. This detection potential has two primary benefits: Firstly, it improves semantic segmentation quality by encouraging consistency between object detections and segmentations.Secondly, it also recalibrates detection

scores.for identifying object instances in the next stage. The output at the semantic segmentation module of the network is denoted as the tensor Q, where $Q_i(l)$ denotes the probability (obtained by applying the softmax function on the network's activations) of pixel i taking on the label $l \in L$.

Instance Segmentation Subnetwork:

At the input to instance segmentation subnetwork,we have two inputs available: The semantic segmentation predictions, Q, for each pixel and label, and a set of object detections. For each input image, we assume that there are D object detections, and that the i^{th} detection is of the form (l_i, s_i, B_i) where $l_i \in L$ is the detected classlabel, $s_i \in [0, 1]$ is the confidence score and B_i is the set of indices of the pixels falling within the detector's bounding box. Note that the number D varies for every input image.

The problem of instance segmentation can then be thought of as assigning every pixel to either a particular object detection, or the background label. This is based on the assumption that every object detection specifies a potential object instance. We define a multinomial random variable, V, at each of the N pixels in the image, and $V = [V_1\ V_2 \ldots V_N]^T$. Each variable at pixel i, V_i, is assigned a label corresponding to its instance. This label set, $\{0, 1, 2, \ldots, D\}$ changes for each image since D, the number of detections, varies for every image (0 is the background label). In the case of instance segmentation of images, the quality of a prediction is invariant to the permutations of instance labelling. For example, in the below figure, labelling the "blue person" as "1" and the "purple person" as "2" is no derent to labelling them as "2" and "1" respectively. This condition is handled by our loss function.

(a) Semantic Segmentation (b) Instance Segmentation

Conditional Random Fields:

A multiclass image labelling problem can be posed as maximum a posteriori (MAP) inference in a CRF defined over pixels. Let's now review conditional random fields used in semantic segmentation and instance segmentation. Take an image I with N pixels, indexed $1, 2, \ldots, N$. In a generic labelling problem, every pixel is assigned a label from a predefined set of labels $\boldsymbol{L} = \{l_1, l_2, \ldots, l_L\}$. Let $\boldsymbol{V} = [V_1\ V_2 \ldots V_N]^T$, where each $V_i \in L$ and $i = 1,2,\ldots,N$. Any particular assignment $\boldsymbol{v}$ to $\boldsymbol{V}$ is thus a solution to the labelling problem.

In the case of semantic segmentation, each pixel, V_i, is assigned a label corresponding to object categories such as "person" and "car". In instance segmentation, the labels are drawn from the product label space of object categories and instance numbers.

Given a graph G where the vertices are from $\{\boldsymbol{V}\}$ and the edges define connections among these variables, the pair $(\boldsymbol{I}, \boldsymbol{V})$ is modelled as a CRF characterised by,

$$Pr(\boldsymbol{V} = \boldsymbol{v}|\boldsymbol{I}) = (1/Z(\boldsymbol{I}))\ exp(-E(\boldsymbol{v}|\boldsymbol{I}))$$

where $E(\boldsymbol{v}|\boldsymbol{I})$ is the energy of labelling $\boldsymbol{v}$ and $Z(\boldsymbol{I})$ is the data dependent partition function. The conditioning on $\boldsymbol{I}$ is dropped hereafter to keep the notation uncluttered. The energy $E(\boldsymbol{v})$ of an assignment is defined using the set of cliques C in the graph G. More specifically,

$$E(\boldsymbol{v}) = \sum_{c \in C} \psi_c(\boldsymbol{v}_c)$$

where $\boldsymbol{v}_c$ is a vector formed by selecting elements of $\boldsymbol{v}$ that correspond to random variables belonging to the clique c and $\psi_c(.)$ is the cost function for the clique c.

Minimising the energy yields the maximum a posteriori (MAP) labelling of the image *i.e.,* the most probable label assignment given the observation (image). When dense pairwise potentials are used in the CRF to obtain higher accuracy, exact inference is impracticable, and one has to resort to an approximate inference method such as mean field inference.

Potentials:

We formulate a Conditional Random Field over our instance variables, V, which consists of unary and pairwise energies. The energy of the assignment v to all the variables, V, is

$$E(V=v)=\sum_i U(v_i)+\sum_{i<j} P(v_i,v_j)$$

The unary energy is a sum of three terms, which take into account the object detection bounding boxes, the initial semantic segmentation and shape information,

$$U(v_i) = -ln[\, w_1 \psi_{Box}(v_i) + w_2\psi_{Global}(v_i)+w_3\psi_{Shape}(v_i)\,]$$

w_1, w_2 *and* w_3 are all weighting coefficients learned via back propagation.

Box Term:

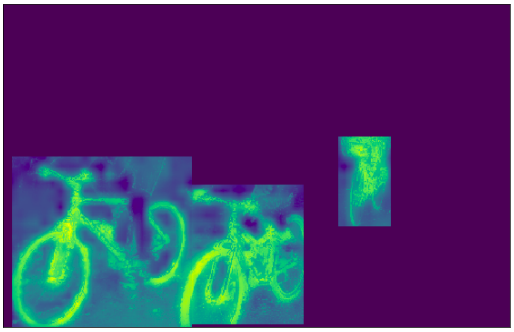
Box term encourages a pixel to be assigned to the instance corresponding to the k^{th} detection if it falls within the detections bounding box. This potential is proportional to the probability of the pixel's semantic class being equal to the detected class $Q_i(l_k)$ and the detection score, s_k.

$$\psi_{Box}(V_i = k) = Q_i(l_k)s_k \qquad if\ i \in B_k$$

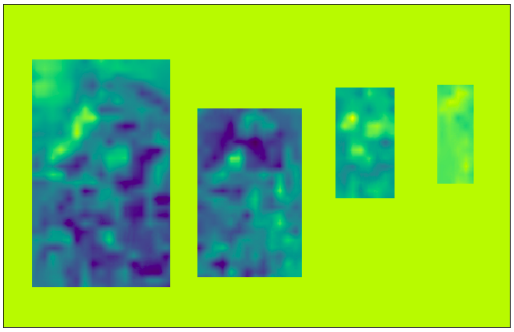
$$= 0 \qquad otherwise$$

This potential performs well when the initial semantic segmentation is good. It is robust to false positive detections, unlike methods which refine bounding boxes since the detections are considered in light of our initial semantic segmentation, Q. Together with the pairwise term , occlusions between objects of the same class can be resolved if there are appearance differences in the different instances.

Results for the box term:



Box term(Cycle)



Box term(Person)

Global Term:

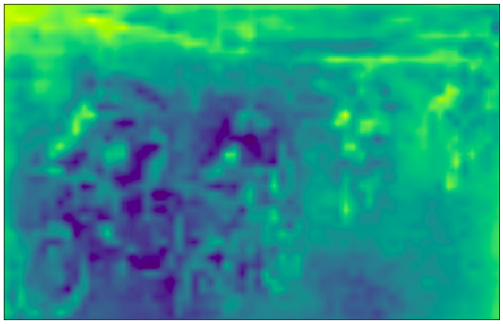
This term does not rely on bounding boxes, but only the segmentation prediction at a particular pixel, Q_i . It encodes the intuition that if we only know there are d possible instances of a particular object class, and have no further localisation information, each instance is equally probable, and this potential is proportional to the semantic segmentation confidence for the detected object class at that pixel:

$$\psi_{Global}(V_i = k) = Q_i(I_k)$$

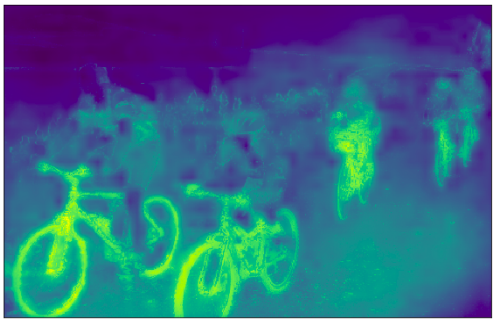
This potential overcomes cases where the bounding box does not cover the entire extent of the object, as it assigns probability mass to a particular instance label throughout all pixels in the image. This is also beneficial during training, as it ensures that the final output is dependent on the segmentation prediction at all pixels in the image, leading to error gradients that are more stable across batches and thus more amenable to backpropagation.

(a) Only Box term (b) Box and Global terms

Results of the global term:



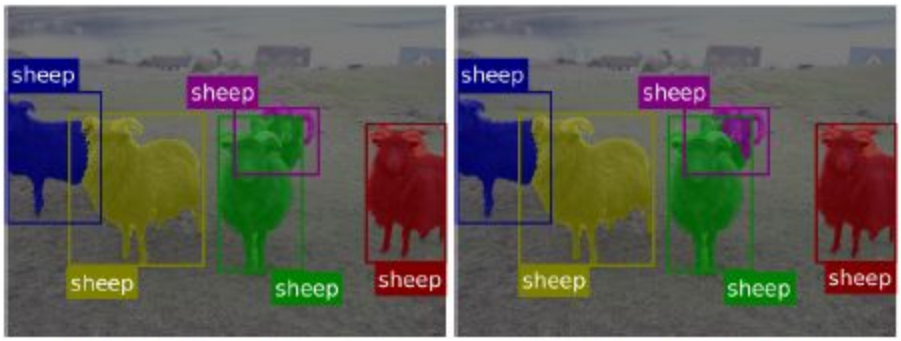
Global term(Cycle)



Global term(Person)

Shape Term:

Shape priors are incorporated to help reason about occlusions involving multiple objects of the same class, which may have minimal appearance variation between them.



(a) Without shape term (b) With Shape term

Given a set of shape templates, T , warp each shape template using bilinear interpolation into $\overline{T}$ so that it matches the dimensions of the k^{th} bounding box, B_k. A shape prior which matches the segmentation prediction for the detected class within the bounding box, $Q_{B_k}(l_k)$, the best according to the normalised cross correlation is selected. The shape prior is element-wise product ($\odot$) between the segmentation unaries and the matched shape prior:

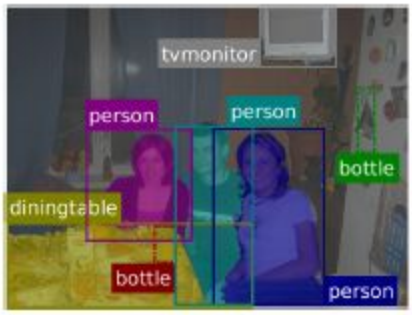
$$t^* = arg\ max_{t \in \overline{T}} \frac{\sum Q_{B_k}(l_k) \odot t}{\| Q_{B_k}(l_k) \| \, \|t\|}$$

$$\psi(V_{B_k} = k) = Q_{B_k}(l_k) \odot t^*$$

Pairwise Term:

The pairwise term consists of densely connected Gaussian potentials and encourages appearance and spatial consistency. They are often able to resolve occlusions based on appearance differences between objects of the same class.

(a) Semantic Segmentation



(b) Instance Segmentation

$$P(v_i,v_j) = \mu(v_i,v_j) \sum_{m=1}^{K} w^{(m)} k^{(m)}(f_i,f_j)$$

$$k^{(m)}(f_i,f_j) = \exp(-0.5\ (f_i,f_j)^T\ \Lambda^{(m)}\ (f_i,f_j))$$

where, f_i, f_j = feature vectors for pixel i and j in arbitrary feature space, $w^{(m)}$ are linear combination weights, μ is a label compatibility function and $\Lambda^{(m)}$ is symmetric, positive-definite precision matrix for each $k^{(m)}$.

Loss Function:

While training for instance segmentation, we have a single loss function which we backpropagate through instance and semantic segmentation modules to update all the parameters. As discussed previously, we need to deal with different permutations of our final labelling which could have the same final result. In this approach we match the original ground truth to our instance segmentation prediction based on the Intersection over Union (IoU) of each instance prediction and ground truth.

More formally, we denote the ground-truth labelling of an image, G, to be a set of r segments, $\{g_1, g_2, \ldots, g_r\}$, where each segment (set of pixels) is an object instance and has an associated semantic class label. Our prediction, which is the output of our network, P, is a set of s segments, $\{p_1, p_2, \ldots, p_s\}$, also where each segment corresponds to an instance label and also has an associated class label. Note that r and s may be different since we may predict greater or fewer instances than actually present. Let M denote the set of all permutations of the ground-truth, G, as different permutations of the ground-truth correspond to the same qualitative result. We define the "matched" ground-truth, G^*, as the permutation of the original ground-truth labelling which maximises the IoU between the prediction, P, and ground truth:

$$G^* = arg\ max_{m \in M}\ IoU(m, P)$$

Once we have the "matched" ground truth, G^* ,for an image, we can apply any loss function to train our network for segmentation. In this case, we use the common cross-entropy loss function. Crucially, we do not need to evaluate all permutations of the ground truth to compute the above equation, since it can be formulated as a maximum-weight bipartite matching problem.The edges in our bipartite graph connect ground-truth and predicted segments. The edge weights are given by the IoU between the ground truth and predicted segments if they share the same semantic class label, and zero otherwise. Leftover segments are matched to "dummy" nodes with zero overlap. Additionally, the ordering of the instances in our network are actually determined by the object detector, which remains static during training. As a result, the ordering of our predictions does not fluctuate

much during training – it only changes in cases where there are multiple detections overlapping an object.

EXPERIMENTAL RESULTS

Experimental details:

Dataset:

- For Object Detection - MS COCO 2017
- For Semantic and Instance Segmentation - PASCAL VOC 2011

Evaluation Metrics:

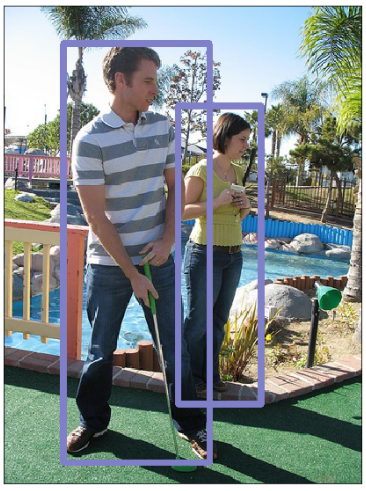
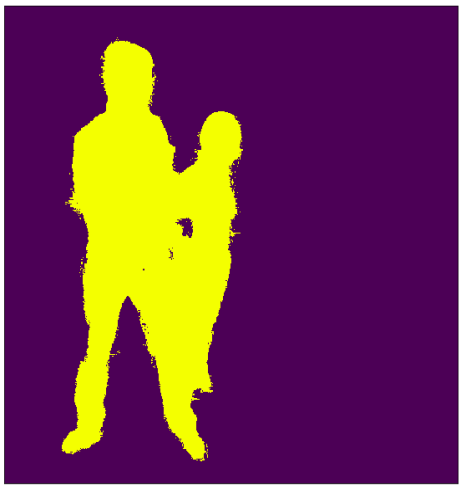
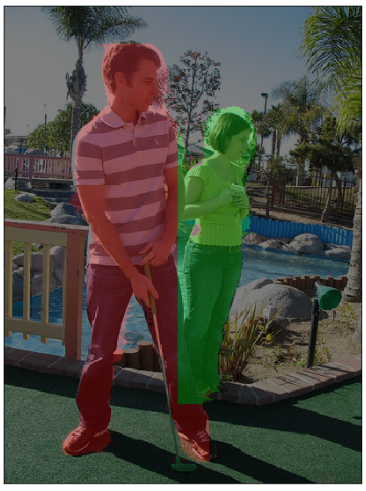
Unary Potential Terms Included	$AP^{0.5}$
Box Term	43.8
Box + Global Term	43.9
Box + Global Term + Shape Term	43.9

Success Cases:







Failure Cases:

REFERENCES

- A. Arnab and P. H. S. Torr. '*Pixelwise instance segmentation with a dynamically instantiated network. In CVPR, 2017*'
- S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. Torr. '*Conditional random fields as recurrent neural networks*'
- D. Weiss and B. Taskar. Scalpel: '*Segmentation cascades with localized priors and efficient learning*'
- Xuming He Richard S. Zemel Miguel A. ´Carreira-Perpin˜an. '*Multiscale Conditional Random Fields for Image Labeling*'