Subspace Alignment Based Domain Adaptation for RCNN Detector

Anant Raj
anantraj@iitk.ac.in
Department of Electrical Engineering,
IIT Kanpur, India

Vinay P. Namboodiri
vinaypn@iitk.ac.in
Department of Computer Science and Engineering,
IIT Kanpur, India

Tinne Tuytelaars
tinne.tuytelaars@esat.kuleuven.be
ESAT, PSI-VISICS,
KU Leuven, Belgium

It has been an underlying assumption behind most of the machine learning algorithms that training and test instances should be sampled from a similar distribution. The difference in resolution, difference in view points, clutter and background are the main reasons which can cause the problem of domain shift. Indeed, the problem is particularly pertinent to the computer vision community due to our reliance on 'standard' challenging datasets. However most of the work in this field revolves around adapting a classifier for the task of object recognition or classification and not much effort has been put to adapt an object detector. In this paper, we propose subspace alignment based domain adaptation of the state of the art RCNN based object detector [2]. In subspace based domain adaptation for objects, we need access to source and target subspaces for the bounding box features . The absence of supervision (labels and bounding boxes are absent) makes the task challenging.

The proposed approach builds upon the previously proposed subspace alignment based method [1] for visual domain adaptation to adapt the RCNN detector [2].

1 Subspace Alignment

Subspace alignment based domain adaptation method consists of learning a transformation matrix M that maps the source subspace to the target one. Suppose, we have labelled source data S and unlabelled target data T. We normalize the data vectors and take separate PCA of the source data vectors and target data vectors. The d eigenvectors for each domain are selected corresponding to the d largest eigenvalues. We consider these eigenvectors as bases for source and target subspaces separately. They are denoted by X_s for source subspace and X_t for target subspace. We use a transformation matrix M to align the source subspace X_s to target subspace X_t. The mathematical formulation to this problem is given by

$$F(M) = \|X_S M - X_T\|_F^2 \qquad M^* = \underset{M}{\operatorname{argmin}}(F(M)). \tag{1}$$

X_s and X_t are matrices containing the d most important eigenvectors for source and target respectively and $\|.\|_F$ is the *Frobenius norm*. The solution of eqn. 1 is $M^* = X_S'X_T$ and hence for the target aligned source coordinate system we get $X_a = X_S X_S' X_T$. Once we get target aligned source co-ordinate system, we project our source data and train the classifier in this frame. While testing, target data is projected on the target subspace and classifier score is calculated.

2 RCNN-detector

Convolutional neural nets (CNN) and other deep learning based approaches have improved the object classification accuracy by a large margin. RCNN [2] uses the CNN framework and bridges the gap between object classification and object detection task. The idea of this work is to see how well the result of convolutional neural network on ImageNet task generalizes for the task of object detection on PASCAL dataset. RCNN consists of three modules. The first module generates selective search windows [3] in an image which is category independent. Second module extracts mid level convolutional neural network features for each proposed region which has been trained earlier on ImageNet dataset. In the third module, SVM classifier is trained by considering all those windows whose overlap with the ground truth bounding box are less then a threshold λ as negative examples, hard negative examples are mined from these negative examples during the training. In testing phase, again 2000 selective search windows is generated per image in fast mode. Each proposal is warped and propagated forward through pre-trained CNN to compute features. Then, for each class, the learned SVM class specific classifier is applied to those extracted features and a score is obtained corresponding to each proposal. Once we get the scores, we decide a threshold and the regions with scores greater than the decided threshold are our possible candidates for a particular object category. In the last step greedy non maximum suppression is applied to obtain desired, accurate and specific bounding box for that object category.

3 Subspace Alignment for Adapting RCNN

In this section we describe our approach to adapt the class specific RCNN -detector. On the basis of background provided in the previous section, we use subspace alignment based domain adaptation over the initial RCNN-detector. Instead of using single subspace for the full source and target data, we postulate that using class-specific different subspaces for different classes to adapt from source to target domain improves the object detection accuracy.

Algorithm 1 Subspace Alignment based Domain Adaptation for RCNN Detector

procedure SA BASED RCNN ADAPTATION(Source Data S,Target Data T)
  for each image $j \in$ *Source* and *Target Image* **do**
    $Windows(j) \leftarrow ComputeSelectiveSearchWindows(j)$
    $feat(j) \leftarrow ComputeCaffeFeat(Windows(j))$
  end for
  $InitRCNNdetector \leftarrow TrainRCNNonSource(SourceData)$
  for each class $i \in Object\ Class$ **do**
    $PosSrc(i) = ()$ and $PosTgt(i) = ()$
    for each image $j \in$ *Source* and *Target Image* **do**
      $ol(j) = ComputOverlap(gTBbox(j,i), Windows(i))$ ▷ For source images
      $PosSrc(i) = \text{Stack}(PosSrc(i), feat(i)(ol(j) \geq \gamma)$ ▷ For source images
      $score(i,j) = runInitRCNNdetector(image(j))$ ▷ For target images
      $PosTgt(i) = \text{Stack}(PosTgt(i), feat(i)(score(i,j) \geq \sigma)$ ▷ For target images
    end for
    $X_{source}(i) \leftarrow PCA(PosSrc(i))$
    $X_{target}(i) \leftarrow PCA(PosTgt(i))$
  end for
  for each class $i \in Object\ Class$ **do**
    $ProjectMat(i) \leftarrow SubspaceAlign(X_{source}(i), X_{target}(i))$
  end for
  $AdaptedRCNNdetector \leftarrow TrainRCNNonSource(ProjectedSrcData)$
  $boxes \leftarrow runAdaptedRCNNdetector(ProjectedTgtData)$
  $predictBbox \leftarrow runNonMaximumSupression(boxes)$
  return $predictBbox$
end procedure

[1] Fernando, Basura, Amaury Habrard, Marc Sebban, and Tinne Tuytelaars. "Unsupervised visual domain adaptation using subspace alignment." In ICCV, 2013 pp. 2960-2967. IEEE, 2013

[2] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014, June). Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, 2014 (pp. 580-587). IEEE

[3] Uijlings, J. R., van de Sande, K. E., Gevers, T., & Smeulders, A. W. (2013). Selective search for object recognition. International journal of computer vision, 104(2), 154-171.