Simultaneous Object Classification and Viewpoint Estimation using Deep Multi-task Convolutional Neural Network

Ahmed J. Afifi[1], Olaf Hellwich[1] and Toufique A. Soomro[2]
[1]*Computer Vision & Remote Sensing, Technische Universität Berlin, Berlin, Germany*
[2]*School of Computing and Mathematics, Charles Sturt University, Bathurst, Australia*

Keywords: Convolutional Neural Networks (CNNs), Multi-task, Object Classification, Viewpoint Estimation, Synthetic Images.

Abstract: Convolutional Neural Networks (CNNs) have shown an impressive performance in many computer vision tasks. Most of the CNN architectures were proposed to solve a single task. This paper proposes a CNN model to tackle the problem of object classification and viewpoint estimation simultaneously, where these problems are opposite in terms of feature representation. While object classification task aims to learn viewpoint invariant features, viewpoint estimation task requires features that capture the variations of the viewpoint for the same object. This study addresses this problem by introducing a multi-task CNN architecture that performs object classification and viewpoint estimation simultaneously. The first part of the CNN is shared between the two tasks, and the second part is two subnetworks to solve each task separately. Synthetic images are used to increase the training dataset to train the proposed model. To evaluate our model, PASCAL3D+ dataset is used to test our proposed model, as it is a challenging dataset for object detection and viewpoint estimation. According to the results, the proposed model performs as a multi-task model, where we can exploit the shared layers to feed their features for different tasks. Moreover, 3D models can be used to render images in different conditions to solve the lack of training data and to enhance the training of the CNNs.

1 INTRODUCTION

Object classification and viewpoint estimation have become popular research topics in computer vision field because of their wide applications. Addressing these two tasks at the same time is beneficial to describe an object under general object recognition task. Object classification is the problem of assigning the correct label to the object in an image. This problem concerns many object classes with different visual instances. For better object understanding, viewpoint estimation is an important step in many applications, such as image retrieval and model matching. Viewpoint estimation is the problem of estimating the view angle with respect to the camera. Also, in scene understanding, it is important to estimate the viewpoint of an object accurately to discover the overall 3D structure of the object and the scene (Penedones et al., 2012) (Su et al., 2015).

Human vision system can recognize different objects of the same class with different viewpoint easily, and it can differentiate between various classes by matching these objects with the correct classes. However, some computerized vision systems can recognize specific objects, but they have troubles in learning and understanding more object categories. Even among some classes, these systems find difficulties in recognizing and classifying some objects because of the changes in lighting conditions, occurrence in different pose, or occurrence in cluttered or occluded environment (Su et al., 2010).

Object classification and viewpoint estimation problems have been studied as separate problems intensively. However, finding a standalone system that is capable of performing both tasks simultaneously is difficult because these tasks have opposite directions in terms of feature representation. For object classification, the system has to learn invariant features with respect to object viewpoint. So it can easily classify the same object that appears in different poses. With regard to estimate the viewpoint of an object, the system has

Afifi, A., Hellwich, O. and Soomro, T.
Simultaneous Object Classification and Viewpoint Estimation using Deep Multi-task Convolutional Neural Network.
DOI: 10.5220/0006544001770184
In *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2018) - Volume 5: VISAPP*, pages 177-184
ISBN: 978-989-758-290-5

to learn a representation that preserves the geometric and the visual information in order to distinguish between different viewpoints of the same object (Zhang et al., 2013).

With the rise of deep learning architectures, many computer vision tasks have been solved using Convolutional Neural Networks (CNNs) such as object recognition and detection (Girshick et al., 2014), segmentation (Shelhamer et al., 2017), and object depth estimation (Afifi and Hellwich, 2016). These problems have been considered either classification problems or regression problems. These architectures have been proposed to solve a single task, and they have shown impressive results. They were pre-trained to perform a specific task and then fine-tuned to perform another task, which is known as transfer learning (Yosinski et al., 2014). Extending these architectures to solve multiple tasks at the same time can be done, but careful design is needed. This means that some layers will be shared for both tasks and some layers will be separated.

The lack of data to train a CNN for a specific task is an irrevocable problem. CNNs need huge number of images to be trained. Fine-tuning a pre-trained CNN can solve the problem of the lack of data if the new task is similar or related to the original task that the CNN has been trained to solve. With the availability of large-scale online 3D models repositories, huge number of images with known viewpoints can be rendered, which can be used for training. In order to make the synthesized images as real ones, the synthesized images can be overlaid with real images as a background image. This step helps the CNN to train on synthesized images, similar to the real images, and to overcome the lack of data issue (Su et al., 2015).

We summarize the contribution of this paper as follows. First, we propose a multi-task CNN architecture that solves jointly object classification and viewpoint estimation tasks. We use a complete synthesized dataset rendered from 3D objects with rich annotations to increase the training dataset and to train the CNN for both tasks. Also, we build a class-dependent subnetwork for viewpoint estimation task that takes care of estimating the viewpoint depends on the object class. Our proposed model showed impressive results in both tasks, and they are comparable to the state-of-the-art results.

2 RELATED WORK

Object classification and viewpoint estimation have been studied in recent years, especially with the evolution of deep learning methods in solving computer vision tasks.

2.1 Object Classification

In (Krizhevsky et al., 2012), the authors proposed the first CNN architecture to solve the problem of object classification. They submitted their results to the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) (Deng et al., 2009). They achieved the top results in the competition. The model was deep consisting of successive convolutional layers, with activation functions and max-pooling layers, and fully connected layers.

Object detection is an important task in computer vision, and can easily be achieved by surrounding the object by a box then classify it. Early work has been achieved by a selective search algorithm (Van de Sande et al., 2011) that generates many region proposals from the input image to recognise them. With respect to the transfer learning concept which introduced in CNN training (Yosinski et al., 2014), many approaches prefer to fine-tune a pre-trained CNN with less training data than ImageNet dataset. In (Girshick et al., 2014), the authors applied the selective research algorithm to generate around 2000 category-independent region proposals and warp them to fine-tune a CNN pre-trained on ImageNet to classify each generated proposal either an object or a background. In (Oquab et al., 2014), the authors fine-tuned a pre-trained CNN trained on ImageNet dataset to compute mid-level image representation from images different from ImageNet dataset and perform object classification on Pascal VOC dataset. Also, (He et al., 2015) obtained state-of-the-art results in object detection and classification by training new fully connected layers on the top of convolutional layers of a network trained previously on ImageNet dataset. They introduce the Spatial Pyramid Pooling layer (SPP) that is flexible enough to handle different scales, sizes, and aspect ratios of the image.

2.2 Viewpoint Estimation

Object orientation is an important geometric feature of the objects in images that can be used for 3D reconstruction. While some previous works dealt with the problem of object viewpoint estimation as a regression problem, we consider this problem as a classification task using CNN. Previous methods focused on estimating object viewpoint of a single object class. They considered simple models of objects without considering the large intra-class

variations. Also, they didn't generalize their methods to handle different object categories because the dataset annotation is insufficient.

Recently, PASCAL3D+ dataset (Xiang et al., 2014) has been introduced as a challenging dataset for object detection and pose estimation. It augments 12 rigid objects with 3D annotations. Most related works use this dataset to evaluate their models on object detection and viewpoint estimation. The output is considered to be correct if the object detection part is correct (the bounding box overlap is larger than 50%) and the viewpoint estimation part is correct. R-CNN (Girshick et al., 2014) and Fast R-CNN (Girshick, 2015) are mostly used as ready detectors to detect the objects first and then estimate the viewpoint. In (Tulsiani and Malik, 2015), the authors consider the viewpoint estimation as a classification problem and train a CNN to predict the viewpoint. To evaluate their model, they use R-CNN to detect the objects and the detected regions are used to estimate the viewpoint. Also, (Poirson et al., 2016) solve the problem of detection and pose estimation in a single fast shot. They combine the detection and the pose estimation at the same level by extending the fast SSD detector (Liu et al., 2016) to estimate object pose at the same time.

To overcome the scarcity of training data to train the CNN for solving the problem of viewpoint estimation, (Su et al., 2015) propose to use synthetic images in training. They render images from 3D online model repositories and mix them with real images for training. We adopt this method to train our model for both object classification and viewpoint estimation.

In sum, object classification and viewpoint estimation tasks have been studied and the proposed methods have attained good results. However, the two tasks are considered separately and separated models are created for each task. Conversely, we consider the two tasks by proposing a new multi-task CNN architecture that performs object classification and viewpoint estimation simultaneously. We also train our model using a synthesized image dataset and test it using a real image dataset.

3 DEEP MULTI-TASK CNN ARCHITECTURE

In this section, we describe our proposed multi-task CNN architecture to solve object classification task and viewpoint estimation task simultaneously. As mentioned before, these two tasks have opposite feature representation requirements. On one hand, the extracted features should be viewpoint invariant to classify the object correctly. On the other hand, viewpoint features should preserve the geometric and the visual features to distinguish between different viewpoints of the same object.

3.1 CNN Architecture

We adopt the well-known CNN architecture introduced by (Krizhevsky et al., 2012) and extend it to solve our problem. This architecture consists of five consecutive convolutional layers followed by three fully connected layers, and it was trained to classify 1K object classes.

So as to build a multi-task CNN model capable of performing two tasks at the same time, we have to decide whether the layers should be shared or separated. Also, we have to decide where to branch the network into two subnetworks. To solve this issue, we propose a multi-task architecture with shared and separated layers to perform as a multi-task model as shown in Figure 1. This architecture contains five convolutional layers that are shared between both tasks. After the fifth convolutional block, the model branches into two subnetworks, one for each task. Each branch consists of three fully connected layers.

The loss function L that is used to train the proposed model has a classification term and a viewpoint term. Formally, it is defined as:

$$L(W) = \lambda_c \,.\, loss_c(x, l^c) + \lambda_{vp} \,.\, loss_{vp}(x, l^{vp}) \qquad (1)$$

where $loss_c$ and $loss_{vp}$ are a softmax function of the object classification task and the viewpoint estimation task, respectively. x is the input image, and l^c and l^{vp} are the class label and the viewpoint label, respectively. λ_c and λ_{vp} are parameters to balance the training process between the two tasks. W is the CNN weights to be learned and optimized. We apply max-pooling after the first, second, and fifth convolutional layers. Max-pooling layers are used to reduce the computation time and to control the overfitting.

In (Yosinski et al., 2014), the authors demonstrate that early layers in CNNs extract generic features, while the last ones are original-dataset-specific layers. Features from early layers can be utilized as general features for different tasks. The last layers extract specific features that help solving a specific task. In our proposed model, the convolutional blocks extract the generic features and

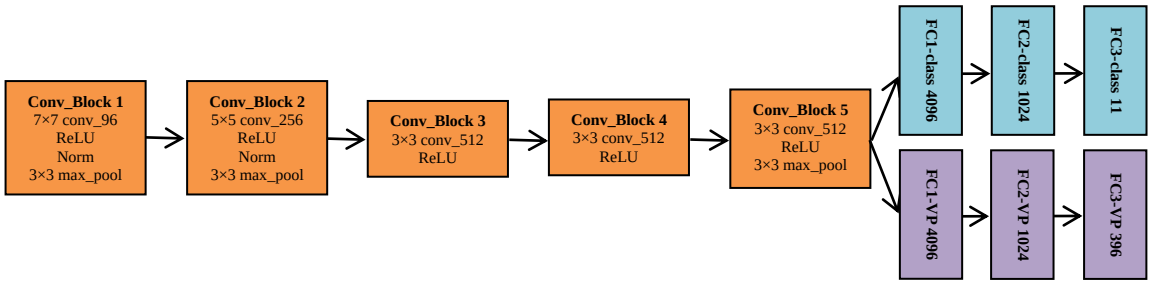

Figure 1: The CNN Architecture. ReLU is the activation function we use. Max-pooling is used in the first, second, and fifth convolutional blocks. Fully connected viewpoint output layer is an object category dependent layer. Fully Connected layers (FC) are followed by a dropout layer with rate 0.5.

the fully connected layers are task-specific layers. So, the network branches after the fifth convolutional block. The new subnetworks contain rich and specific features that represent the objects for a specific task as shown in Figure 1.

Concerning the viewpoint subnetwork, the first two fully connected layers are used for all classes. The last fully connected layer is a class-dependent layer. That is each class has its own fully connected layer to estimate the viewpoint. We use a separated fully connected layer for each class because viewpoint estimation depends on the geometric properties of the classes, and there is a huge geometric variation between the classes. So, applying a model trained on a specific class will not perform well on another class. And, creating a separated subnetwork for each class and train it independently is a naïve solution.

3.2 Implementation Details

We use MatConvNet (Vedaldi and Lenc, 2015), a MATLAB toolbox implementing CNNs for computer vision application, to implement and evaluate our proposed model. The weights of the shared layers are initialized using the corresponding weights in VGG-m model (Chatfield et al., 2014), which was pre-trained on the ILSVRC data for image classification. The weights of the classification subnetwork layers are initialized from the same network, while the viewpoint subnetwork weights were initialized randomly. We fine-tune the subnetworks using back-propagation. Stochastic gradient descent (SGD) method is used to optimize our network with the following settings: the momentum is set to 0.9, and the weight decay is set to 10^{-5}. The learning rate is initialized to 10^{-3} and is decreased by 10 when the validation error doesn't change.

With reference to the object classification and viewpoint estimation, we train our model on 11 object classes which are introduced in PASCAL3D+ dataset (Xiang et al., 2014). For the viewpoint estimation task, it is known that the nature of the viewpoint is continuous, and many research works deal with this problem as a regression problem (Schwarz et al., 2015). However, we consider the task as a classification problem. More specifically, we focus on estimating the azimuth angle and we divide the viewpoint range into 36 classes (10 angles in each class).

Regarding the training data, CNNs are always hungry and need a massive amount of data for training. With the availability of large-scale online 3D model repositories (Chang et al., 2015), we use these 3D models to render object images with different orientations. The rendering process helps to introduce more images to train the CNN for object classification and viewpoint estimation. We can control the rendering process and generate a huge number of images. Rendered images provide a reasonable number of images for the CNN to be trained well (Su et al., 2015). Also, we performed data augmentation on the rendered images which resulted in increasing the training data. We also overlaid the rendered images with real images as background to guide the network towards convergence and to avoid wrong classification.

4 EXPERIMENTAL RESULTS

In this section, we will present the experimental results for object classification and viewpoint estimation, respectively. Before that, we will introduce the test dataset which we will use.

PASCAL3D+ (Xiang et al., 2014) is a well-known challenging dataset for object detection and viewpoint estimation captured in the wild. It contains 12 rigid categories of PASCAL VOC 2012

(Everingham et al., 2010) with rich 3D annotations. That is, each object is annotated with its viewpoint (azimuth angle, elevation angle, and distance from the camera) and bounding box values surrounding the object in the image. Furthermore, more images were added from ImageNet (Deng et al., 2009) for each category with rich annotations, and we use them to evaluate our model on object classification task.

4.1 Object Classification Results

We test our model on ImageNet dataset for object classification on the same object classes that are introduced in PASCAL3D+ dataset (11 object classes). Table 1 shows the performance of our proposed model. We use the mean Average Precision (mAP) as a metric to evaluate our model.

From Table 1, we can notice that our proposed model can classify the objects accurately. This model was trained on synthesized images and tested on real images. The results show that we can use synthesized images to train CNNs and enhance the performance of the trained model. We got 93.1% classification mean Average Precision when we consider the maximum output value from the CNN.

We also conduct another experiment to evaluate the performance of our model using PASCAL VOC 2012 val dataset on the same object classes introduced in PASCAL3D+ dataset with rich annotations. This dataset is a challenging dataset because the images were captured in the wild, and each image contains many objects of different classes. We use the ground-truth bounding box to extract the object from the input image and resize it to fit the CNN model input size. Table 1 shows the performance of our model on PASCAL VOC 2012 val dataset. We notice that the chair and the table classes record low accuracy. The reason behind this is that they appear mostly together in the same image, and when we extract the object, it appears as a cluttered or occluded object. We can conclude that the proposed model performs well for object classification.

To compare our proposed architecture performance to other methods, we use PASCAL VOC 2007 test dataset as most of the literature uses this dataset. Our proposed model solves object classification and viewpoint estimation problem simultaneously. We compared our results to (Wu et al., 2015), (Oquab et al., 2014), and (Razavian et al., 2014). In (Wu et al., 2015), the authors proposed a deep learning framework in weakly supervised settings that can classify multiple objects in a single image and perform image annotation. The authors in (Oquab et al., 2014) proposed a method to exploit the image representations learned by CNNs trained on large-scale annotated dataset to other recognition task. They used the layers trained on ImageNet dataset to extract mid-level features from PASCAL VOC dataset and trained new layers for object classification problem. That is, they have applied the transfer learning concept to exploit the pre-trained layers to extract generic features and train new layers on different dataset for the same task. (Razavian et al., 2014) used the features extracted from OverFeat network (Sermanet et al., 2013) as generic features to solve many object recognition tasks, such as object classification and scene recognition. For each task, they selected a suitable dataset according to the task. After that, a linear SVM classifier is applied on the extracted features from the network. We have to point out that the previously mentioned works used the whole 20 object classes in PASCAL VOC 2007 test dataset to test their proposed methods. However, we just train and test our proposed model on the object classes introduced in PASCAL3D+ dataset, which are 11 object classes. Also, the images we used in training are synthetic images to overcome the problem of the lack of data. Table 2 shows the comparison between the results reported in the previous works and our results. It is clear that our proposed method outperforms the previous work in object classification task.

4.2 Viewpoint Estimation Results

First, we present a comparison between two different choices of CNN architecture with respect to the output layer, either class-specific or general outputs, and we show that the output layer should be class dependent because of the geometric differences between the object classes. Then, we compare our proposed network results with some previous works that address the same problem.

4.2.1 Comparison of Different Viewpoint Estimation Networks

We compare two different choices of models for viewpoint estimation with respect to the output layer. The first model is the same proposed model with respect to the viewpoint subnetwork as shown in Figure 1, where the last fully connected layer is a class specific layer. That is, for each class we train a separate layer. We denote this model by *specific_model*. The second model shares the last

Table 1: Object Classification Performance on ImageNet dataset and PASCAL VOC 2012 val dataset.

Dataset	aero	bicycle	boat	bus	car	chair	d.table	mbike	sofa	train	tv	mAP
ImageNet	99.0	94.3	98.4	94.5	97.4	72.1	97.7	90.8	86.3	95.1	97.9	**93.1**
VOC 12 val	89.6	81.7	79.1	81.0	77.5	76.1	65.6	78.9	58.0	78.6	85.8	**77.5**

Table 2: Object Classification results and comparison with other methods on PASCAL VOC 2007 test dataset.

Method	aero	bicycle	boat	bus	car	chair	d.table	mbike	sofa	train	tv	mAP
Wu et al., 2015	**93.5**	83.4	83.6	81.6	86.6	54.5	53.8	79.0	63.7	**91.5**	80.4	77.4
Oquab et al., 2014	88.5	81.5	82.0	75.5	**90.1**	61.6	67.3	80.0	58.0	90.4	77.9	77.5
Razavian et al., 2014	90.1	84.4	**84.1**	73.4	86.7	61.3	69.6	80.0	67.3	89.1	74.9	78.3
Ours	90.4	**86.7**	76.9	**84.3**	87.5	**77.2**	**73.3**	**81.0**	**67.4**	77.1	**87.5**	**80.9**

fully connected layer between all classes. So, it doesn't care about the object class when the model estimates the object viewpoint. We denote this model by *general_model*. We train both proposed models on synthetic images and test them on PASCAL VOC 2012 val dataset. We use the ground-truth bounding box to extract the object from the image during testing, and we use the Average Precision (AP) as a metric to compare the performance of the models. We perform the experiment on 4 different viewpoint categorizations as introduced in (Xiang et al., 2014). Figure 2 shows the results of both models, and we can clearly notice that the class specific layers at the end of the subnetwork perform better and more accurate than when using the same fully connected layer for all classes. This is because the variety of the geometry information between different classes, and even between the objects in the same class.

4.2.2 Comparison to Previous Methods

We conduct an experiment to evaluate the performance of our model on object viewpoint estimation and compare it with other models. We use PASCAL VOC 2012 val dataset as most of the previous methods used it to evaluate their models.

As introduced in PASCAL3D+, Average Viewpoint Precision (AVP) metric (Xiang et al., 2014) is used to evaluate object detection and viewpoint estimation. That is, the output is considered to be correct if and only if the bounding box overlap that detects the object is larger than 50% and the viewpoint is correct. As our problem is to solve only the viewpoint estimation task, we use R-CNN (Girshick et al., 2014) detector to generate the bounding box. Other methods use either their own detectors or other proposed detectors like Fast R-CNN (Girshick, 2014). Table 3 shows the detailed comparison between our method and other previous methods handling the same problem.

We compare our model with the following models: DPM-VOC+VP (Pepik et al., 2012) which uses a modified version of DPM to predict viewpoint, Render for CNN (Su et al., 2015) which uses R-CNN for object detection, and Viewpoints & Keypoints (Tulsiani and Malik, 2015). The authors in (Poirson et al., 2016) proposed a fast model that detect the object using SSD detector and estimate its pose, and they achieved comparable results with the state-of-the-art. In (Massa et al., 2016), the authors achieved the state-of-the-art results of viewpoint estimation on PASCAL3D+. They use VGG16 (Simonyan and Zisserman, 2014) model for viewpoint estimation and Fast R-CNN (Girshick, 2015) for detection. Our model achieved a comparable accuracy to the state-of-the-art models, and we can conclude that we got a model that can classify the object and estimate the viewpoint at the same time.

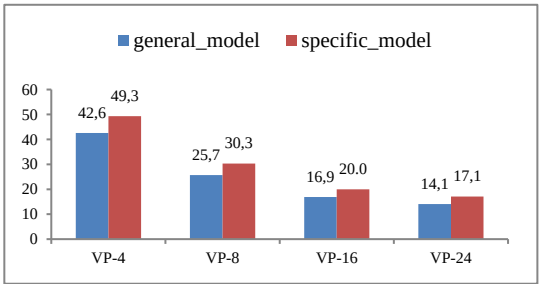

Figure 2: Comparison of Different Viewpoint Discretization between different Viewpoint Estimation Models.

Table 3: Viewpoint estimation results and comparisons with the state-of-the-art methods on PASCAL VOC 2012 val dataset. The methods are referenced as follow: DPM-VOC+VP (Pepik et al., 2012), Render for CNN (Su et al., 2015), Vps & Kps (Tulsiani and Malik, 2015), Fast SSD (Poirson et al., 2016), and Craft. CNN (Massa et al., 2016).

Methods	aero	bicycle	boat	bus	car	chair	d.table	mbike	sofa	train	tv	Avg.
Joint Object Detection and Viewpoint Estimation (**4 View AVP**)												
DPM-VOC+VP	37.4	43.9	0.3	48.6	36.9	6.1	2.1	31.8	11.8	11.1	32.2	23.8
Render for CNN	54.0	50.5	15.1	57.1	41.8	15.7	18.6	50.8	28.4	46.1	58.2	39.7
Vps & Kps	63.1	59.4	23.0	69.8	55.2	25.1	24.3	61.1	43.8	59.4	55.4	49.1
Fast SSD	64.6	62.1	26.8	70.0	51.4	11.3	40.7	62.7	40.6	65.9	61.3	50.6
Ours	58.4	60.8	29.1	62.1	50.3	37.6	41.5	59.1	55.6	55.9	51.3	**51.1**
Craft. CNN	70.3	67.0	36.7	75.4	58.3	21.4	34.5	71.5	46.0	64.3	63.4	**55.4**
Joint Object Detection and Viewpoint Estimation (**8 View AVP**)												
DPM-VOC+VP	28.6	40.3	0.2	38.0	36.6	9.4	2.6	32.0	11.0	9.8	28.6	21.5
Render for CNN	44.5	41.1	10.1	48.0	36.6	13.7	15.1	39.9	26.8	39.1	46.5	32.9
Vps & Kps	57.5	54.8	18.9	59.4	51.5	24.7	20.5	59.5	43.7	53.3	45.6	44.5
Fast SSD	58.7	56.4	19.9	62.4	45.2	10.6	34.7	58.6	38.8	61.2	49.7	45.1
Ours	49.6	55.9	22.2	60.8	44.7	32.2	31.2	55.4	46.1	53.1	50.3	**45.6**
Craft. CNN	66.0	62.5	31.1	68.7	55.7	19.2	31.9	64.0	44.7	61.8	58.0	**51.3**
Joint Object Detection and Viewpoint Estimation (**16 View AVP**)												
DPM-VOC+VP	15.9	22.9	0.3	49.0	29.6	6.1	2.3	16.7	7.1	20.2	19.9	17.3
Render for CNN	27.5	25.8	6.5	45.5	29.7	8.5	12.0	31.4	17.7	29.7	31.4	24.2
Vps & Kps	46.6	42.0	12.7	64.6	42.7	20.8	18.5	38.8	33.5	42.5	32.9	36.0
Fast SSD	46.1	39.6	13.6	56.0	36.8	6.4	23.5	41.8	27.0	38.8	36.4	33.3
Ours	31.9	40.3	13.5	55.9	37.8	25.8	24.6	41.7	41.0	47.2	44.2	**36.7**
Craft. CNN	51.4	43.0	23.5	68.9	46.3	15.2	29.3	49.4	35.6	47.0	37.3	**40.6**
Joint Object Detection and Viewpoint Estimation (**24 View AVP**)												
DPM-VOC+VP	9.7	16.7	2.2	42.1	24.6	4.2	2.1	10.5	4.1	20.7	12.9	13.6
Render for CNN	21.5	22.0	4.1	38.6	25.5	7.4	11.0	24.4	15.0	28.0	19.8	19.8
Vps & Kps	37.0	33.4	10.0	54.1	40.0	17.5	19.9	34.3	28.9	43.9	22.7	31.1
Fast SSD	43.2	29.4	9.2	54.7	35.7	5.5	23.0	30.3	27.6	44.1	34.3	28.8
Ours	26.4	30.7	11.2	53.9	34.1	23.2	23.3	33.3	37.3	45.0	40.2	**32.6**
Craft. CNN	43.2	39.4	16.8	61.0	44.2	13.5	29.4	37.5	33.5	46.6	32.5	**36.1**

5 CONCLUSIONS

In this paper, we tackled the problem of object classification and viewpoint estimation simultaneously. We presented a new multi-task CNN architecture that has shared layers performing as features extraction layers for both tasks and separated subnetworks for each task. Owing to the opposite nature of the two tasks, the branching is necessary. Object classification task requires viewpoint invariant features, while viewpoint estimation task requires capturing the variations of the viewpoint for different objects of different classes. We also trained our network on synthesized images and this helped us in solving the problem of the lack of data problem. Our results showed that the proposed model has high accuracy on both tasks and is comparable to the state-of-the-art methods.

REFERENCES

Afifi, A. J. and Hellwich, O., 2016. Object Depth Estimation from a Single Image Using Fully Convolutional Neural Network. *In Digital Image Computing: Techniques and Applications (DICTA), 2016 International Conference on (pp. 1-7). IEEE.*

Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H. and Xiao, J., 2015. Shapenet: *An information-rich 3d model repository.* arXiv preprint arXiv:1512.03012.

Chatfield, K., Simonyan, K., Vedaldi, A. and Zisserman, A., 2014. Return of the devil in the details: *Delving*

deep into convolutional nets. arXiv preprint arXiv:1405.3531

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K. and Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database. *In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on (pp. 248-255). IEEE.*

Everingham, M., Van Gool, L., Williams, C.K., Winn, J. and Zisserman, A., 2010. The pascal visual object classes (voc) challenge. *International journal of computer vision, 88(2), pp.303-338.*

Girshick, R., Donahue, J., Darrell, T. and Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. *In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 580-587).*

Girshick, R., 2015. Fast r-cnn. *In Proceedings of the IEEE international conference on computer vision (pp. 1440-1448).*

He, K., Zhang, X., Ren, S. and Sun, J., 2015. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence, 37(9), pp.1904-1916.*

Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. *In Advances in neural information processing systems (pp. 1097-1105).*

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y. and Berg, A.C., 2016. Ssd: Single shot multibox detector. *In European conference on computer vision (pp. 21-37).* Springer, Cham.

Massa, F., Marlet, R. and Aubry, M., 2016. Crafting a multi-task CNN for viewpoint estimation. *In BMVC (pp. 1-10).*

Oquab, M., Bottou, L., Laptev, I. and Sivic, J., 2014. Learning and transferring mid-level image representations using convolutional neural networks. *In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1717-1724).*

Penedones, H., Collobert, R., Fleuret, F. and Grangier, D., 2012. *Improving object classification using pose information (No. EPFL-REPORT-192574).* Idiap.

Pepik, B., Stark, M., Gehler, P. and Schiele, B., 2012, June. Teaching 3d geometry to deformable part models. *In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on(pp. 3362-3369).* IEEE.

Poirson, P., Ammirato, P., Fu, C.Y., Liu, W., Kosecka, J. and Berg, A.C., 2016. Fast single shot detection and pose estimation. In 3D Vision (3DV), 2016 *Fourth International Conference on (pp. 676-684).* IEEE.

Schwarz, M., Schulz, H. and Behnke, S., 2015. RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features. *In Robotics and Automation (ICRA), 2015 IEEE International Conference on (pp. 1329-1335).* IEEE.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R. and LeCun, Y., 2013. *Overfeat: Integrated recognition, localization and detection using convolutional networks.* arXiv preprint arXiv: 1312.6229.

Sharif Razavian, A., Azizpour, H., Sullivan, J. and Carlsson, S., 2014. CNN features off-the-shelf: an astounding baseline for recognition. *In Proceedings of the IEEE conference on computer vision and pattern recognition workshops (pp. 806-813).*

Shelhamer, E., Long, J. and Darrell, T., 2017. Fully convolutional networks for semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence, 39(4), pp.640-651.*

Simonyan, K. and Zisserman, A., 2014. *Very deep convolutional networks for large-scale image recognition*. arXiv preprint arXiv:1409.1556.

Su, Y., Allan, M. and Jurie, F., 2010. Improving object classification using semantic attributes. *In BMVC (pp. 1-10).*

Su, H., Qi, C.R., Li, Y. and Guibas, L.J., 2015. Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views. *In Proceedings of the IEEE International Conference on Computer Vision (pp. 2686-2694).*

Tulsiani, S. and Malik, J., 2015. Viewpoints and keypoints. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1510-1519).*

Van de Sande, K.E., Uijlings, J.R., Gevers, T. and Smeulders, A.W., 2011. Segmentation as selective search for object recognition. *In Computer Vision (ICCV), 2011 IEEE International Conference on (pp. 1879-1886).* IEEE.

Vedaldi, A. and Lenc, K., 2015. Matconvnet: Convolutional neural networks for matlab. *In Proceedings of the 23rd ACM international conference on Multimedia (pp. 689-692).* ACM.

Wu, J., Yu, Y., Huang, C. and Yu, K., 2015. Deep multiple instance learning for image classification and auto-annotation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 3460-3469).*

Xiang, Y., Mottaghi, R. and Savarese, S., 2014. Beyond pascal: A benchmark for 3d object detection in the wild. *In Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on(pp. 75-82).* IEEE.

Yosinski, J., Clune, J., Bengio, Y. and Lipson, H., 2014. How transferable are features in deep neural networks?. *In Advances in neural information processing systems (pp. 3320-3328).*

Zhang, H., El-Gaaly, T., Elgammal, A.M. and Jiang, Z., 2013, July. Joint Object and Pose Recognition Using Homeomorphic Manifold Analysis. *In AAAI(Vol. 2, p. 5).*

Zhang, H., El-Gaaly, T., Elgammal, A. and Jiang, Z., 2015. Factorization of view-object manifolds for joint object recognition and pose estimation. *Computer Vision and Image Understanding, 139, pp.89-103.*