Recent Advances on Application of Deep Learning for Recovering Object Pose*

Wanyi Li, *Member, IEEE*, Peng Wang, *Member, IEEE*, Zhengke Qin, Hai Zhou, Hong Qiao, *Senior Member, IEEE*

***Abstract*— Recovering object pose is of great importance to many higher level tasks such as robotic manipulation, scene understanding and augmented reality to name a few. Following the recent major breakthroughs in many computer vision tasks made by the deep learning, intensive research to experiment with it also in the task of recovering object pose is conducting. This paper aims to review the state-of-the-art progress on deep learning based pose estimation methods. Firstly, we introduce some popular datasets together with their relevant attributes. Secondly, the deep learning based pose estimation methods are summarized and categorized, and detailed descriptions of representative methods are provided, and their pros and cons are examined. Thirdly, evaluation protocol and comparable performance of reviewed approaches are given. Finally, we highlight the advantages of deep learning based pose estimation methods and provide insights for future.**

I. INTRODUCTION

Recovering accurate object pose, especially 6-DoF pose, is a difficult but important computer vision task with a long research history and has many practical applications such as robotic manipulation (like Amazon Picking Challenge), scene understanding and augmented reality to name a few. A large number of methods have been developed for recovering object pose from photometric images, range images, and more recently, from registered color/depth images (RGB-D data). However, there still remain several challenges to address, such as foreground occlusions, background clutter, large scale and pose changes and multi-instance objects.

Deep Learning is a new area of Machine Learning research which is about learning multiple levels of representation and abstraction that help to make sense of data such as images, sound, and text [1, 2]. These methods have made major breakthrough in many computer vision tasks such as image classification [3], object detection [4], object tracking [5] etc. Thus, deep learning is expected to improve significantly the performance of object pose estimation too. Following the recent impressive performance boost in many computer vision fields brought by the deep learning, intensive research to experiment with it also in the task of recovering object pose is conducting and some deep learning based 3D object detection and pose estimation approaches have been proposed.

The goal of this paper is to review the state-of-the-art progress on deep learning based pose estimation methods. More thorough reviews on general object pose estimation can be found in [6-8]. Specifically, this paper introduces some important datasets in Section II. Section III first introduces several categorization factors, then summarizes and categorizes the major deep learning based pose estimation methods. The evaluation protocol to evaluate object pose estimation methods and comparisons of the performance of surveyed methods are given in Section IV. Finally, Section V concludes this paper by presenting a brief discussion on potential future research directions in this research area.

II. DATASETS

Many databases have been built to test various algorithms. A set of popular 2D camera image and 2.5d RGB-D image databases are listed together with their major attributes in TABLE I. . The symbol '-' denotes that the corresponding item is not reported. In TABLE I. , RGB-D data type means registered color/depth images. 3D models are also simultaneously provided in some datasets.

III. DEEP LEARNING BASED POSE ESTIMATION METHODS

In this section, we first introduce 4 factors that will be used later for categorization of pose estimation methods.

- Data type

The data types used includes RGB image, Range image, RGB-D data, i.e., registered color/depth images which are usually captured by Kinect or Kinect-style sensor, LiDAR data and 3D model (or CAD data).

- Architecture

Several deep architectures have been used in deep learning based pose estimation methods, such as 2D Convolutional Neural Network (2D CNN), 3D CNN, Deep Belief Networks (DBN), hierarchical sparse coding and auto-encoder.

- DoF of pose

The DoFs of estimated pose usually are 1 DoF, 2 DoF, 3 DoF or 6 DoF.

- Open source or not

This factor indicates whether the source code of method is available on line. If it is open source, the link will be provided.

*This work was supported by National Natural Science Foundation of China (61401463, 61379097, 61210009) and China Postdoctoral Science Foundation funded project (2015M572500)

Wanyi Li, Peng Wang and Zhengke Qin are with the Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing, CO 100190, China(corresponding author: Peng Wang, phone: 010-82544535-824; email:wanyi.li@ia.ac.cn, peng_wang@ia.ac.cn, qinzhengke@hotmail.com)

Hai Zhou is with Research Center of Laser Fusion, China Academy of Engineering Physics, Mianyang, CO 621900, China.

Hong Qiao is with State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, CO 100190, China.

The main difference of the surveyed methods is the used deep architecture, thus methods are explained based on the used deep architecture. We introduce the methods in chronological order. A summary of deep learning based object pose estimation methods and their categorization according to factors mentioned above is presented in TABLE II.

TABLE I. SOME IMPORTANT IMAGE DATASETS FOR RECOVERING OBJECT POSE

No.	Name and reference	Data type	Acquisition	#Objects	#Images	Annotations	Challenge factors	3D model Available
1	ACCV3D Dataset/ Linemod Dataset [9]	RGB-D	Kinect	15	Over 1100/ object	6 DoF pose	Texture-less; Heavy clutter	Yes
2	Multi-Instance Dataset [10]	RGB-D	Kinect style	6	~800 /object	6 DoF pose	Multiple instances ; Cluttered background; Partial occlusions	Yes
3	Bin-Picking Dataset[a] [11]	RGB-D	Kinect style	2	~100/per scene	6 DoF pose	Multiple instances; Severe Occlusions; Scanning window contains parts from multiple instances	Yes
4	6-DOF Pose Tracking Dataset [12]	RGB-D	Kinect	3	Six sequences	6 DoF pose	Fast-moving objects; Object occlusions	Yes
5	20 Objects Light Dataset [13]	RGB-D	Kinect	20	10000	6 DoF pose	Realistic noise patterns; Challenging lighting conditions	Yes
6	UoB Highly Occluded Object Challenge (UoB-HOOC) [14]	3D point cloud (in addition to RGB)	Kinect v2	up to 20	-	6 DoF pose	High levels of occlusions; Numerous symmetric objects	Yes
7	Desk3D [15]	RGB-D	Kinect	6	1000	6 DoF pose	Clutter ; Occlusion	Yes
8	T-LESS [16]	RGB-D	Primesense Carmine 1.09; Kinect v2; Canon Digital IXUS 950 IS	30	~1800	6 DoF pose	Relatively small objects often very similar in shape and color; Significant clutter and occlusions	Yes
9	Rutgers APC RGB-D Dataset [b] [17]	RGB-D	Kinect	24	10,368 depth and RGB registered images	6 DOF pose	Clutter; variety in size, shape, texture and transparency	Yes
10	"Shelf & Tote" Benchmark Dataset [c] [18]	RGB-D	RealSense F200 RGB-D camera	39	7,281 images	6 DOF object poses and segmentation labels	Cluttered environments, self-occlusion, sensor noise, and a large variety of objects	Yes
11	RGBD Object Dataset [19]	RGB-D	Kinectstyle	300 objects organized into 51 categories	250,000	1 DoF pose (rotation angle)	-	No
12	KITTI Dataset [20]	RGB with Velodyne point cloud	RGB camera; Velodyne laser sensor	~80	7481 training images and 7518 test images	2D / 3D bounding box , orientation in bird's eye view	Different size; Occlusion; Truncation	No
13	Pascal3D [21]	RGB	RGB camera	12 rigid categories, 3,000 object instances per category	> 20,000	3D pose	A great amount of variability; Much more object instances; Occlusions; Clutter	Yes
14	ObjectNet3D [22]	RGB	RGB camera	44,147 3D shapes organized into 100 categories	90,127 images, 44,147 3D shapes	3D pose annotation	Variability of object categories and geometry	Yes
15	EPFL car dataset [23]	RGB	RGB camera	1	20 sequences of cars	Rotation angle	-	No

a. The first fully annotated bin-picking dataset, found in industrial setups. b. Objects of which are used during the first Amazon Picking Challenge (APC). c. Used in APC 2016.

A. 2D CNN based methods

CNNs have greatly advanced the performance in the large-scale visual recognition challenge (ILSVRC2012) [3]. The key point to the success of CNNs is that their ability to learn rich feature representations as opposed to hand-crafted features used in previous image classification methods. The 2D CNN based pose estimation methods are conceptually very similar to feature learning in images for image classification. These methods utilize CNN to learning feature from RGB images or from RGB-D images, where depth is treated as an additional input channel.

Yu et al. [24] utilize the Max-pooling Convolutional Neural Network (MPCNN) for pose estimation from RGB image, in which different poses of objects are assigned as different classes.

Based on CNN, Yang et al. [25] propose a novel deep architecture, called Auto-masking Neural Network (ANN) for simultaneous object detection and viewpoint estimation. ANN contains multiple CNNs and a mask layer and thus can learn to select the most discriminative object parts across different viewpoints from training images. Besides, a new method is presented to estimate the continuous viewpoint, which makes the estimation more accurate. Experimental results show ANN outperforms the state-of-the-art algorithms. This work only considers the estimation of viewpoint in horizontal directions without considering object tilt angles.

To represent objects in an RGB-D scene with corresponding 3D models, [26] first detects and segments out object instances using CNN based features [27], and then use a CNN to estimate coarse pose. This CNN is trained using pixel surface normals in images containing renderings of synthetic objects. With the coarse pose estimate and the inferred pixel support , a small number of prototypical models are aligned to the data, and the model that fits best are placed into the scene. A 48% relative improvement in performance at the task of 3D detection over the current state-of-the-art [28] is achieved, while being an order of magnitude faster.

Su et al. [29] propose a scalable and over fit resistant image synthesis pipeline, together with a novel CNN specifically tailored for the viewpoint estimation task. Experimental results show that the viewpoint estimation from the pipeline can significantly outperform state-of-the-art methods on PASCAL 3D benchmark [21].

Wohlhart et al. [30] train a CNN to compute descriptors of object views that efficiently capture both the object identity and 3D pose by enforcing simple similarity and dissimilarity constraints between the descriptors. The learnt descriptors can generalize to unseen objects. The method can work with either RGB or RGB-D images and outperforms state-of-the-art methods on the ACCV3D dataset [9]. However the input of the method is only regions containing the objects to be detected instead of the whole test images.

Inspired by Wohlhart et al. [30], Kehl et al. [31] propose to use regressed descriptors of locally-sampled RGB-D patches to perform reliable 3D object detection and pose estimation under clutter and occlusion. For regression, they employ a convolutional auto-encoder that has been trained on a large collection of random local patches. During testing, scene patch descriptors are matched against a database of synthetic model view patches and cast 6D object votes which are subsequently filtered to refined hypotheses. Evaluation on three datasets shows that this method delivers robust detection results that compete with and surpass the state-of-the-art while being scalable in the number of objects, and generalizes well to previous unseen input data.

To cope with the difficulties including occlusion and complicated sensor noise in 6D pose estimation in RGB-D image, Krull et al. [32] train a CNN to compare rendered and observed images. The CNN is used to describe the posterior density of a particular object pose and is trained with the maximum likelihood paradigm. Compared to state-of-the-art, this method achieves a significant improvement on two different datasets which include a total of eleven objects, cluttered background, and heavy occlusion. However, it is noted that but their work is restricted to single object instances and does not deal with the combinatorics of multi-object pose estimation.

Bonde et al. [33] present a novel deep architecture termed templateNet for depth based object instance recognition. This architecture exploits prior knowledge of an object's shape to sparsify the feature maps by using an intermediate template layer. This regularizes the network and improves its performance without additional parameterization.

Schwarz et al. [34] propose to address RGB-D object recognition and pose estimation with transfer learning from deep CNNs that are pre-trained for image categorization and provide a rich, semantically meaningful feature set. This method incorporates depth information, which the CNN was not trained with, by rendering objects from a canonical perspective and colorizing the depth channel according to distance from the object center. Evaluation on the RGB-D object dataset [19] demonstrates that the generated feature set naturally separates classes and instances well and retains pose manifolds. This method outperforms state-of-the-art on a number of subtasks and can yield superior results when only little training data is available.

Elhoseiny et al. [35] study how CNN architectures can be adapted to the task of joint object categorization and pose estimation. The authors investigate and analyze the layers of various CNN models and extensively compare between them with the goal of discovering how the layers of distributed representations within CNNs represent object pose information and how this contradicts with object category representations. Extensive experiment on two recent large and challenging multi-view datasets show that the proposed method achieves better than the state-of-the-art.

Zeng et al. [18] present the vision system of Team MIT Princeton's 3rd- and 4th-place entry in the 2016 Amazon Picking Challenge (APC2016). The proposed framework leverages multi-view RGB-D data and data-driven, self-supervised deep learning to reliably estimate the 6D poses of objects. A deep convolutional neural network is trained to segment RGBD point clouds captured from multiple views into different objects. Then pre-scanned 3D models of the identified objects are aligned to the segmented point clouds to estimate the 6D pose of each object. It worth noting that the vision system of two team of three winners in APC2016 are also based on deep learning methods [36]. Their performance data of vision system are not publicly available currently yet.

The 2D CNN based pose estimation methods have the advantage of utilizing efficiently the mature architecture and pre-trained models in object recognition. This further helps understanding how the CNNs unify the different requirement for view-invariant representation learning for recognition and pose-information-kept representation learning for estimating object pose.

B. 3D CNN based methods

The 3D CNN based methods differ from those based on 2D CNN in that the 3D CNN based methods employ a fully volumetric representation and process 3D data directly, resulting in a richer and more discriminative representation.

Wu et al. [37] propose to represent a geometric 3D shape as a probability distribution of binary variables on a 3D voxel grid, using a Convolutional Deep Belief Network and apply it to RGB-D object recognition, among other tasks.

Song et al., 2015 [38] introduce a 3D ConvNet formulation, named Deep Sliding Shapes, of which the input is a 3D volumetric scene from a RGB-D image and the output is 3D object bounding boxes. This method utilizes a 3D Region Proposal Network (RPN) to learn objectness from geometric shapes and uses a joint Object Recognition Network (ORN) to extract geometric features in 3D and color features in 2D for the recognition part. Experiments show that the algorithm outperforms the state-of-the-art [28] by 13.8 in mAP and is 200x faster than the original Sliding Shapes.

Maturana et al. [39] propose an architecture named VoxNet for real-time 3D object recognition by integrating a volumetric Occupancy Grid representation with a supervised 3D Convolutional Neural Network (3D CNN). Evaluation on publicly available benchmarks using LiDAR, RGBD, and CAD data shows that VoxNet achieves accuracy beyond the state of the art while labeling hundreds of instances per second.

Based on the VoxNet [39] , Sedaghat et al. [40] propose Orientation-boosted Voxel Nets for 3D recognition in which the network is forced to predict the coarse pose of the object in addition to the class label. This approach yields better results than the VoxNet [39].

The 3D CNN based methods not only can fully utilize a rich source of 3D information from range sensors such as LiDAR and RGBD cameras that can aid in 3D object recognition, but also can efficiently deal with large amounts of point cloud data. However, such approaches cannot work on high resolution 3D data, as the computational complexity is a cubic function of the voxel grid resolution. Besides the 3D CNN based methods for estimating 6D pose exactly are quite scarce and worth studying further.

C. Other methods

The deep architectures used in methods of this category can be hierarchical sparse coding Deep Belief Networks (DBN), Autoencoder and Field Probing Neural Networks (FPNN).

Bo et al. [41] introduce hierarchical matching pursuit (HMP) for feature learning from RGB-D data. HMP uses sparse coding to learn codebooks at each layer in an unsupervised way and builds hierarchical feature representations from the learned codebooks in conjunction with orthogonal matching pursuit, spatial pooling and contrast normalization. Extensive experiments on various datasets indicate that the features learned with this approach enable superior object recognition results using linear support vector machines.

Following [24], Liang et al. [42] apply a new Deep Belief Networks consisting of two traditional DBNs together for the same task, i.e. 3D object recognition and pose estimation, and get better result.

Doumanoglou et al. [11] propose a complete framework for 6 DoF object detection in crowded scenes, comprising of an architecture based on Sparse Auto-encoders for unsupervised feature learning, 6 DoF object pose estimation using Hough Forests and a technique based on Hough Forests for predicting the next-best-view. Two additional challenging datasets are provided. One is related to domestic environments and the other depicts a bin-picking scenario mostly found in industrial settings. Evaluation on challenging datasets shows superior results.

Li et al. [43] represent 3D spaces as volumetric fields, and propose a novel design that employs field probing filters to efficiently extract features from them. The proposed Field Probing Neural Networks (FPNN) is significantly more efficient than 3DCNNs, while providing state-of-the-art performance, on classification tasks for 3D object recognition benchmark datasets.

TABLE II. SUMMARY OF DEEP LEARNING BASED POSE ESTIMATION METHODS

No	Method	Data type	Architecture	DoF of pose	Open source
2D CNN based methods					
1	Yu et al., 2013 [24]	RGB image	2D CNN	1	No
2	Yang et al.,2014[25]	RGB image	2D CNN	1	No
3	Gupta et al., 2014 [26]	RGB-D data	2D CNN	6	No
4	Su et al., 2015 [29]	RGB image	2D CNN	1	Yes, https://shapenet.cs.stanford.edu/projects/RenderForCNN/
5	Wohlhart et al., 2015 [30]	RGB/RGB-D/Range image	2D CNN	6	Yes, https://cvarlab.icg.tugraz.at/projects/3d_object_detection/
6	Kehl et al. [31]	RGB-D data	Convolutional auto-encoder	6	No
7	Krull et al., 2015[32]	RGB-D data	2D CNN	6	No
8	Bonde et al., 2015 [33]	Range image	2D CNN	6	No
9	Schwarz et al. , 2015 [34]	RGB-D data	2D CNN	1	No
10	Elhoseiny et al., 2016 [35]	RGB image	2D CNN	1, 3	No
11	Zeng et al., 2016 [18]	RGB-D data	2D CNN	6	Yes, https://github.com/andyzeng/apc-vision-toolbox

No	Method	Data type	Architecture	DoF of pose	Open source
3D CNN based methods					
12	Wu et al., 2015 [37]	RGB-D data/3D model	3D Convolutional DBN	6	Yes. http://3dshapenets.cs.princeton.edu/
13	Song et al., 2015 [38]	RGB-D data	3D CNN	0	Yes, https://github.com/shurans/DeepSlidingShape
14	Maturana et al., 2015 [39]	LiDAR data/ RGB-D data / CAD data	3D CNN	0	Yes, https://github.com/dimatura/voxnet
15	Sedaghat et al., 2016 [40]	LiDAR data/ RGB-D data / CAD data	3D CNN	0	No.
Other methods					
16	Bo et al., 2014 [41]	RGB-D data	Sparse coding	1	Yes, http://research.cs.washington.edu/istc/lfb/software/hmp/index.htm
17	Liang et al., 2015[42]	RGB image	DBN	1	No
18	Doumanoglou et al., 2015 [11]	RGB-D data	Sparse Autoencoder	6	No
19	Li et al. 2016 [43]	CAD data	Field probing based neural networks (FPNN)	0	Yes, https://github.com/yangyanli/FPNN

IV. Evaluation and Performance

A. Evaluation Protocol

There are a series of evaluation criteria which are frequently used to assess the performance of a pose estimation system. Some of them are used for performance comparison in TABLE III. .

(1) Evaluation criteria for methods with full pose (6 DoF)

Recognition rate. Recognition rate is the percentage of scenes where the object was successfully found, as well as the pose is correctly estimated. In order to assess whether a 6 DoF pose solution is correct, the following three metrics are used: 1. The Average Distance (AD) criterion (Hinterstoisser et al. [9]): The average distance between all vertices in the 3D model in the estimated pose and the ground truth pose. A pose is considered correct, when this average distance is below 10% of the object diameter. 2. 5cm, 5deg (Shotton et al. [44]): A pose is considered correct when the translational error is below 5cm and the rotational error is below 5deg. 3. The Intersection Over Union (IOU) criterion: The 2D axis aligned bounding boxes in the estimated pose and ground truth pose is first calculated. Then calculate the IOU of the bounding boxes and a pose is considered correct, when this value is above a threshold (to be defined).

Location(L) accuracy and **Location+Pose(L+P) accuracy** [15]: Localization is considered to be correct if the predicted center is within a fixed radius ($\frac{\max(w,d,h)}{3}$) of the ground truth position. Pose classification is considered to be correct if the predicted pose class (largest pose probability) or template is either the closest or second closest quantized pose to the ground truth.

(2) Evaluation criteria for methods with part pose

Mean Precision of Pose Estimation (MPPE) [45]. MPPE is the average classification accuracy of multiple pose classes and equivalents to the average over the diagonal of the viewpoint class confusion matrix.

Average Viewpoint Precision (AVP) [21]. AVP is the average precision with a modified true positive definition, requiring both 2D detection and viewpoint estimation to be correct.

$\mathbf{Acc_{\pi/6}}$ and MedErr [46]. $Acc_{\pi/6}$ measures accuracy (the higher the better) and MedErr measures error (the lower the better) based on geodesic distance over the manifold of rotation matrices.

(3) Evaluation criteria for methods without pose

Average Precision (AP) [47]. The 2D object detection performance is evaluated by the widely used criterion Average Precision (AP) established in the Pascal VOC challenge [47].

3D detection Average Precision (AP) [28]. The evaluation scheme of 3D object detection is similar to 2D's. For 3D object, we assume the boxes are aligned with gravity direction, and calculate the 3D bounding box overlapping ratio. The threshold for 3D is set to be 0.25.

B. Performance

To compare the reviewed methods further, we list the performance of all reviewed methods in TABLE III. . The listed attributes include performance (i.e. recognition rate, accuracy etc.) , time cost, software platform and hardware. The symbol '-' denotes that the corresponding item is not reported.

Compared performances in TABLE III. show the following key points. Firstly, as expected, most of them achieve good performance, some of them [11, 25, 26, 29, 30, 32-35, 39, 41, 43] even outperform state-of-the-art methods. Secondly, some methods [11, 25, 34, 39, 43] achieve start-of-the-art accuracy while run in real time. Thirdly, there is a large space to improve and time cost is also a big problem if pose estimation method is design for real time application.

TABLE III. Performance of Deep Learning Based Pose Estimation Methods

No	Method	Performance	Time cost	Software	Hardware
2D CNN based methods					
1	Yu et al., 2013[24]	Self-built dataset with 5 objects, recognition rate 94.5%	-	-	-
2	Yang et al.,2014[25]	(1)Detection and Discrete Viewpoint Estimation. 3D Object Classes car dataset [48], AP [47] / MPPE [45]: 99.9% / 97.9%; EPFL car dataset [23], AP [47] / MPPE [45]: 99.6 %/ 71.4% (2) Continuous Viewpoint Estimation. EPFL car dataset [23], Median Angular Error (MAE) / Mean Angular Error (MnAE): 3.3 / 24.1.	> 1fps , 300 × 400pixels	-	A PC with a NVIDIA GTX 670 GPU
3	Gupta et al., 2014 [26]	NYUV2 dataset [49], 3D detection AP [28] (depth only): 57.6%, 3D detection AP [28] (depth + img): 58.5%	40 seconds CPU + 30 seconds on a GPU per categories per image	C++, Caffe	-
4	Su et al., 2015 [29]	Dataset: PASCAL 3D [21]. (1)Simultaneous object detection and viewpoint estimation , AVP [21]: 39.7%. (2)Viewpoint estimation with ground truth bounding box, $Acc_{\pi/6} = 0.76,\ \text{MedErr} = 11.7°$	-	Caffe, Python+Matlab	-
5	Wohlhart et al., 2015[30]	ACCV3D dataset [9]. Recognition rate with pose error below 20° : 96.2%; Recognition rate when the pose is ignored: 99.8%	-	Theano+Python	-
6	Kehl et al. [31]	(1) Multi-instance dataset from Tejani et al, F1:0.747 (2) Linemod dataset, F1: around 0.9 for each object (3) Challenge dataset, F1: 0.956	1.7fps Sub-linear with the number of objects	-	-
7	Krull et al., 2015 [32]	Occlusion Dataset from [13] and [9], recognition rate with AD [9]: 72.98%; 6-DOF Pose Tracking Dataset [12], recognition rate with AD [9]: 56.74%	One training cycle took 9min 46sec; Time cost of testing: -	-	Intel(R) Core (TM) i7-3820 CPU at 3.60GHz GeForce GTX 660 GPU
8	Bonde et al., 2015 [33]	(1). Location+Pose accuracies [15] on non-occluded scenes of Desk3D [15]: 80.13% ; (2). 9 large objects (bounding box > 1000cc) of ACCV3D dataset [9], Location(L) accuracy [15] :92.5, Location+Pose(L+P) accuracy[15]: 78.01%.	-	-	-
9	Schwarz et al. , 2015 [34]	RGB-D object dataset [19]. Category Accuracy: 89.4%; Instance Accuracy: 94.1%. avgPose(I)a: 42.8°	5fps	-	Intel Core i7-4800MQ CPU @ 2.7 GHz +NVidia GeForce GT 730M
10	Elhoseiny et al., 2016 [35]	(1) RGB-D object dataset [19]. Category recognition accuracy: 97.14%; Pose (AAAI) [50]:79.3%; (2) Pascal3D dataset [21]. Category recognition accuracy: 83%; Pose (AAAI) [50]:73.53%;	-	-	-
11	Zeng et al., 2016 [18]	"Shelf & Tote" Benchmark Dataset [18], recognition rate with rotations within 15°: 49.9%,recognition rate with translations within 5cm:66.1%.	20 seconds or less per frame	C++, Matlab	An Intel E3-1241 CPU 3.5 GHz and an NVIDIA GTX 1080
3D CNN based methods					
12	Wu et al., 2015 [37]	Accuracy for View-based 2.5D Recognition on NYUV2 dataset[49]: 0.579.	-	Matlab	-
13	Song et al., 2015 [38]	NYUV2 dataset[49], 3D detection AP [28] (depth only): 67.8%, 3D detection AP [28] (depth + img): 72.3%.	(1)training, RPN : 10 hours; ORN: 17 hours (2) testing: RPN : 5.62s/image; ORN: 13.93s/image	Matlab	NVIDIA K40 GPU
14	Maturana et al., 2015 [39]	Avg F1 on Sydney Object dataset: 0.72; Avg acc on ModelNet10 [37]: 0.92; Avg acc on ModelNet40 [37]: 0.83; Avg acc on NYUV2 dataset[49]: 0.71.	(1) training: around 6 to 12 hours; (2) Testing: 2ms-6ms/ object instance	Lasagne ,C++ and Python	K40 GPU
15	Sedaghat et al., 2016 [40]	Avg F1 on Sydney Object dataset: 0.778; Avg acc on ModelNet10 [37]: 0.938; Avg acc on NYUV2 dataset[49]: 0.763. Avg acc on KITTI Dataset [20]: 0.937	-	-	-
Other methods					
16	Bo et al., 2014 [41]	RGB-D object dataset [19]. AvgPose(I)ª:44.8° .	-	Matlab	-
17	Liang et al., 2015[42]	Dataset of [24], recognition rate :97.3%; Self-built dataset with 4 objects, recognition rate :98.3%, Pose estimation correct rate: 97.5%	-	-	-
18	Doumanoglou et al., 2015 [11]	Recognition rate with AD [9] on the 20 Objects Light Dataset [13]: 89.1%. Recognition rate with AD [9] on the Multi-Instance Dataset [10] :80.3%.	Training: train a tree in 90 mins; Testing: 6-9secs/frame	-	i7 CPU

No	Method	Performance	Time cost	Software	Hardware
19	Li et al. 2016 [43]	Avg acc on ModelNet40 [37]: 0.884	Running time of field probing layers at all grid resolution: <3ms	Caffe, C++, Python	Nvidia GeForce GTX TITAN GPU

a. Mean pose error of test images that were assigned the correct instance.

V. Conclusion and Future Directions

In this paper we survey the recent progress on deep learning based object pose estimation methods. After introducing some important datasets and evaluation metric for object pose estimation, we summarize and categorize the major algorithms in this research area, including 2D CNN based methods, 3D CNN based methods and other deep architecture based methods. We also compare the performance of surveyed methods.

Although various deep learning based pose estimation methods have been invented in recent years, no single one is robust and fast enough to deal with all situations due to occlusions, clutter, large scale and pose changes and multi-instance objects. Thus there are much work to conduct further. Some of them are stated as follows.

A promising solution is to combine the merits of both objectness proposal of visual attention mechanism and 2D CNN based descriptor learning similar to [30]. In which objectness proposal generates candidate object region fast and learned descriptor is used to match the candidate region and object view. An iterative closest point (ICP) step can be used as a following pose refinement step.

Another promising direction is to construct 3D CNN for both RGB-D object recognition and pose estimation. Since the existing 3D CNN based methods usually only for 3D object recognition, shape retrieval and 3D model classification without pose information. There are much work to do in adapting 3D CNN for recovering object pose. Combining 3D object proposal and 3D CNN feature learning worth trying also.

Finally, exploring the pre-trained models successfully used in object recognition into object pose estimation task is also an exciting topic. Such as adapting pre-trained model to a unify model for simultaneous object recognition and pose estimation, in which some layers extract view-invariant feature for recognition and some other layers extract pose-information-kept representation for estimating object pose. Besides, using two separate model learnt at different stages for simultaneous object recognition and pose estimation is also good solution worth trying. A model learns view-invariant representation for categorization while the other model learns pose-information-kept representation for recovering pose.

References

[1] "Deep Learning Tutorial," *LISA lab, University of Montreal*, 2015.

[2] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436-444, 2015.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, 2012.

[4] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580-587.

[5] N. Wang and D. Y. Yeung, "Learning a deep compact image representation for visual tracking," in *Advances in Neural Information Processing Systems*, 2013, pp. 809-817.

[6] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, and J. Wan, "3D object recognition in cluttered scenes with local surface features: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, pp. 2270-2287, 2014.

[7] A. Aldoma, Z. C. Marton, F. Tombari, W. Wohlkinger, C. Potthast, B. Zeisl, *et al.*, "Tutorial: Point Cloud Library: Three-Dimensional Object Recognition and 6 DOF Pose Estimation," *IEEE Robotics & Automation Magazine*, vol. 19, pp. 80-91, 2012.

[8] S. Savarese and L. Fei-Fei, "Multi-view object categorization and pose estimation," in *Computer Vision*, ed: Springer, 2010, pp. 205-231.

[9] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, *et al.*, "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," in *Asian Conference on Computer Vision*, 2012.

[10] A. Tejani, D. Tang, R. Kouskouridas, and T. K. Kim, "Latent-Class Hough Forests for 3D Object Detection and Pose Estimation," in *ECCV*, 2014, pp. 462-477.

[11] A. Doumanoglou, R. Kouskouridas, S. Malassiotis, and T.-K. Kim, "Recovering 6D Object Pose and Predicting Next-Best-View in the Crowd," *arXiv preprint arXiv:1512.07506*, 2015.

[12] A. Krull, F. Michel, E. Brachmann, S. Gumhold, S. Ihrke, and C. Rother, "6-DOF Model Based Tracking via Object Coordinate Regression," in *ACCV*, 2014, pp. 384-399.

[13] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother, "Learning 6D object pose estimation using 3D object coordinates," in *ECCV*, 2014, pp. 151-173.

[14] K. Walas and A. Leonardis. (2016). *UoB Highly Occluded Object Challenge (UoB-HOOC)*. Available: http://www.cs.bham.ac.uk/research/projects/uob-hooc/

[15] U. Bonde, V. Badrinarayanan, and R. Cipolla, "Robust Instance Recognition in Presence of Occlusion and Clutter," in *European Conference on Computer Vision (ECCV)*, 2014.

[16] T. Hodaň, P. Haluza, Š. Obdržálek, J. Matas, M. Lourakis, and X. Zabulis. (2016). *T-LESS An RGB-D Dataset and Evaluation Protocol for Detection and 6D Pose Estimation of Textureless Objects*. Available: http://cmp.felk.cvut.cz/t-less/

[17] C. Rennie, R. Shome, K. E. Bekris, and A. F. D. Souza, "A Dataset for Improved RGBD-Based Object Detection and Pose Estimation for Warehouse Pick-and-Place," *IEEE Robotics & Automation Letters*, vol. 1, pp. 1179-1185, 2015.

[18] A. Zeng, K.-T. Yu, S. Song, D. Suo, J. Walker, E., A. Rodriguez, *et al.*, "Multi-view Self-supervised Deep Learning for 6D Pose Estimation in the Amazon Picking Challenge," *ArXiv e-prints*, September 2016.

[19] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view rgb-d object dataset," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, 2011, pp. 1817-1824.

[20] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012, pp. 3354-3361.

[21] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond PASCAL: A benchmark for 3D object detection in the wild," in *IEEE Winter Conference on Applications of Computer Vision*, 2014, pp. 75-82.

[22] Y. Xiang, W. Kim, W. Chen, J. Ji, C. Choy, H. Su, *et al.*, "ObjectNet3D: A Large Scale Database for 3D Object Recognition," in *European Conference on Computer Vision (ECCV)*, 2016, pp. 160-176.

[23] M. Ozuysal, V. Lepetit, and P. Fua, "Pose estimation for category specific multiview object localization," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2009, pp. 778-785.

[24] J. Yu, K. Weng, G. Liang, and G. Xie, "A vision-based robotic grasping system using deep learning for 3D object recognition and pose estimation," in *Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on*, 2013, pp. 1175-1180.
[25] L. Yang, J. Liu, and X. Tang, "Object detection and viewpoint estimation with auto-masking neural network," in *European Conference on Computer Vision*, 2014, pp. 441-455.
[26] S. Gupta, P. Arbeláez, R. Girshick, and J. Malik, "Aligning 3D models to RGB-D images of cluttered scenes," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4731-4740.
[27] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning Rich Features from RGB-D Images for Object Detection and Segmentation," in *European Conference on Computer Vision (ECCV)*, 2014, pp. 345-360.
[28] S. Song and J. Xiao, "Sliding Shapes for 3D Object Detection in Depth Images," in *European Conference on Computer Vision (ECCV)*, 2014, pp. 634-651.
[29] H. Su, C. R. Qi, Y. Li, and L. J. Guibas, "Render for CNN: Viewpoint Estimation in Images Using CNNs Trained with Rendered 3D Model Views," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2686-2694.
[30] P. Wohlhart and V. Lepetit, "Learning descriptors for object recognition and 3D pose estimation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3109-3118.
[31] W. Kehl, F. Milletari, F. Tombari, S. Ilic, and N. Navab, "Deep Learning of Local RGB-D Patches for 3D Object Detection and 6D Pose Estimation," in *European Conference on Computer Vision (ECCV)*, 2016.
[32] A. Krull, E. Brachmann, F. Michel, M. Y. Yang, S. Gumhold, and C. Rother, "Learning Analysis-by-Synthesis for 6D Pose Estimation in RGB-D Images," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 954-962.
[33] U. Bonde, V. Badrinarayanan, R. Cipolla, and M.-T. Pham, "TemplateNet for Depth-Based Object Instance Recognition," *arXiv preprint arXiv:1511.03244*, 2015.
[34] M. Schwarz, H. Schulz, and S. Behnke, "RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 1329-1335.
[35] M. Elhoseiny, T. El-Gaaly, A. Bakry, and A. Elgammal, "A Comparative Analysis and Study of Multiview CNN Models for Joint Object Categorization and Pose Estimation," in *Proceedings of The 33rd International Conference on Machine Learning*, 2016, pp. 888-897.
[36] N. Corporation. (2016). *Pickup Artist: GPU-Powered Robot Wins Amazon Warehouse Challenge*. Available: https://blogs.nvidia.com/blog/2016/08/12/warehouse-automation-amazon-picking-challenge/
[37] W. Zhirong, S. Song, A. Khosla, Y. Fisher, Z. Linguang, T. Xiaoou, *et al.*, "3D ShapeNets: A deep representation for volumetric shapes," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1912-1920.
[38] S. Song and J. Xiao, "Deep Sliding Shapes for amodal 3D object detection in RGB-D images," *arXiv preprint arXiv:1511.02300*, 2015.
[39] D. Maturana and S. Scherer, "VoxNet: A 3D Convolutional Neural Network for real-time object recognition," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, 2015, pp. 922-928.
[40] N. Sedaghat, M. Zolfaghari, and T. Brox, "Orientation-boosted Voxel Nets for 3D Object Recognition," *arXiv preprint arXiv:1604.03351*, 2016.
[41] L. Bo, X. Ren, and D. Fox, "Learning hierarchical sparse features for RGB-(D) object recognition," *The International Journal of Robotics Research*, vol. 33, pp. 581-599, 2014.
[42] D. Liang, K. Weng, C. Wang, G. Liang, H. Chen, and X. Wu, "A 3D object recognition and pose estimation system using deep learning method," in *2014 4th IEEE International Conference on Information Science and Technology*, 2014, pp. 401-404.
[43] Y. Li, S. Pirk, H. Su, C. R. Qi, and L. J. Guibas, "FPNN: Field Probing Neural Networks for 3D Data," *arXiv preprint arXiv:1605.06240*, 2016.
[44] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, "Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013, pp. 2930-2937.
[45] B. Pepik, M. Stark, P. Gehler, and B. Schiele, "Teaching 3D geometry to deformable part models," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012, pp. 3362-3369.
[46] S. Tulsiani and J. Malik, "Viewpoints and keypoints," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1510-1519.
[47] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision*, vol. 88, pp. 303-338, 2010.
[48] S. Savarese and F.-F. Li, "3D generic object categorization, localization and pose estimation," in *2007 IEEE 11th International Conference on Computer Vision*, 2007, pp. 1-8.
[49] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor Segmentation and Support Inference from RGBD Images," in *12th European Conference on Computer Vision (ECCV)*, Berlin, Heidelberg, 2012, pp. 746-760.
[50] K. Lai, L. Bo, X. Ren, and D. Fox, "A Scalable Tree-Based Approach for Joint Object and Pose Recognition," in *AAAI*, 2011.