applied sciences

MDPI

Article

3D Wireframe Modeling and Viewpoint Estimation for Multi-Class Objects Combining Deep Neural Network and Deformable Model Matching

Xiaoyuan Ren, Libing Jiang *, Xiaoan Tang and Weichun Liu

College of Electronic Science, National University of Defense Technology; Changsha HN 731, China, renxiaoyuan10@nudt.edu.cn (X.R.); xatang@nudt.edu.cn (X.T.); liuweichun17@nudt.edu.cn (W.L.)

* Correspondence: jianglibing@nudt.edu.cn; Tel.: +86 15581641708

Received: 25 February 2019; Accepted: 8 May 2019; Published: 14 May 2019

Featured Application: This research is a useful exploration to extend the generalization of deep learning in 3D modeling and viewpoint estimation.

Abstract: The accuracy of 3D viewpoint and shape estimation from 2D images has been greatly improved by machine learning, especially deep learning technology such as the convolution neural network (CNN). However, current methods are always valid only for one specific category and have exhibited poor performance when generalized to other categories, which means that multiple detectors or networks are needed for multi-class object image cases. In this paper, we propose a method with strong generalization ability, which incorporates only one CNN with deformable model matching processing for the 3D viewpoint and the shape estimation of multi-class object image cases. The CNN is utilized to detect keypoints of the potential object from the image, while a deformable model matching stage is designed to conduct 3D wireframe modeling and viewpoint estimation simultaneously with the support of the detected keypoints. Besides, parameter estimation by deformable model matching processing has robust fault-tolerance to the keypoint detection results containing mistaken keypoints. The proposed method is evaluated on Pascal3D+ dataset. Experiments show that the proposed method performs well in both parameter estimation accuracy and the multi-class objects generalization. This research is a useful exploration to extend the generalization of deep learning in specific tasks.

Keywords: 3D vision; viewpoint estimation; wireframe modeling; deformable model

1. Introduction

Estimating the 3D geometry of an object from a single image is an important but challenging task in computer vision [1]. Recent years have witnessed an emerging trend towards analyzing the 3D viewpoint and shape instead of merely providing 2D bounding boxes. Previously, 3D primitives were fitted with the image to obtain viewpoint and shape parameters [2,3]. While these primitives can provide detailed descriptions of objects, robustly matching them to real-world images was proven to be difficult. Recently, the developments of machine learning especially deep neural networks such as the convolution neural network (CNN), have contributed greatly to this field. Despite the good performance gained by these methods, they share a common limitation: Each network or detector is trained for only one specific category target generally.

For the 3D shape and the viewpoint estimation problem, most of the existing methods are interested in reconstructing 3D model for category-specific objects [4–10]. In general, the deformable model of the specific category, such as wireframe and mesh, is matched with the image to estimate the shape and the viewpoint. Although current methods using deep learning technology can output the shape and viewpoint parameters in an end-to-end way and perform well in accuracy, most of them are limited in one specific category and exhibit poor performance when generalized to other categories. Consequently, multiple detectors or networks are needed for multi-class cases, which significantly raises the training costs. Considering the fact that the sofas and the chairs have similar legs, as well as that the cars and the bicycles all contain wheels, it is worth pointing out that different object categories do share rich compositional similarities. Consequently, Zhou [11] took advantage of this characteristic and constructed the Starmap network in 2018, which can extract the keypoints for multi-class objects. However, this network cannot be directly adopted here due to its intrinsic disadvantage of lacking semantic information for the extracted keypoints.

In this paper, the problem of 3D viewpoint and shape estimation for multi-class objects from a single image is further investigated. Instead of producing multiple detectors or networks just like category-specific methods, the proposed approach uses only one keypoint detection network incorporating it with the deformable model matching processing. Firstly, the keypoint detection network for multi-class objects is trained. Keypoint locations of multi-class objects can be obtained through this network, but unlike the category-specific methods, the semantic meaning of each detected keypoint is not provided. In the following, these extracted keypoints are utilized and explored for deformable model matching, which can be divided into two stages: model selection and model validation. In the first stage, the extracted 3D keypoints are utilized to match with different deformable models corresponding to multi-classes objects for the selection of candidate deformable models. In the second stage, these candidate models are further screened and validated by the matching with the extracted 2D keypoints, which provides semantic meaning to each keypoint, and can be used to conduct 3D wireframe modeling and viewpoint estimation simultaneously.

The main contributions of our research are as follows: Firstly, only one keypoint detection network is adpoted for multi-class objects in the proposed method, which can not only reduce the training cost, but also capture similarity across different categories. Secondly, the deformable model matching processing is introduced to utilize and supplement results obtained from the network. Besides, parameter estimation by the deformable model has robust fault-tolerance to the mistaken keypoints. In conclusion, the method proposed combines the advantage of deep learning and priori model. Compared with methods that depend only on a deep network, the proposed approach has better generalization performance. This paper explores how to extend the generalization of deep learning in a specific task.

2. Related Work

2.1. Viewpoint and Shape Estimation

In earlier days of computer vision, single objects, as well as entire scenes, were represented by simple primitives, such as polyhedra [2] and generalized cylinders [3]. These approaches provided rich descriptions of objects and could estimate the viewpoint and the shape parameters, but robustly matching them to cluttered real-world images proved to be difficult at the time. With the advent of computers and advances in machine learning, it has become feasible to detect objects and their parts robustly. Currently, vision-based methods can be broadly classified into 2D image-based and 3D model-based techniques [4].

Image recognition techniques are employed by 2D image-based methods to attempt to directly restore pose information from the single image [5,6]. These methods usually work by a set of trained model-views taken in a range around the known model with different locations and viewpoints, which always suffer from intra-category variations. Pose estimation using 3D model-based methods usually require a priori 3D model of the object, and the holistic cost function is defined when the 3D deformable model is fitted to the image features. Pepik [7] extended the deformable part model (DPM) to 3D, and Xiang [8] introduced a separate DPM component corresponding to each viewpoint.

These methods can estimate the viewpoint and the shape parameters simultaneously with different representation including wireframe and 3D mesh. Lately, estimation accuracy and utility have been greatly improved in the deep learning era. The single image 3D interpreter network (3D-INN) [9] presented a sophisticated convolutional neural network (CNN) architecture to estimate a 3D skeleton containing viewpoint and shape information. Chi Li obtained the 3D object structure by a deep CNN architecture with domain knowledge in hidden layers [10]. However, these methods rely on category-specific keypoint annotation and are not generalizable. When dealing with multi-class objects, different networks need to be trained separately, which ignores inter-category structure similarities and raises training costs significantly. This paper promotes a method to attain a generalization ability.

2.2. Keypoint Detection

Wireframe representation is a concise structure modeling form with strong description ability, which can preserve the structural properties in 3D modeling. In order to extract the wireframe from the image, keypoint detection is necessary. Researchers have made significant progress in detecting keypoints. The traditional way is to train the classifier with the hand-craft feature that is used in DPM [12]. Recently, there have been several attempts to apply CNN to detect keypoints.

Toshev [13] trained a deep neural network for 2D human pose regression. Xiang Yu optimized deformation coefficients based on the principal component analysis (PCA) representation of 2D keypoints to achieve state-of-the-art performance on the face and human body [14]. Despite the good performance of these approaches, they share a common limitation: Each keypoint is only trained for a specific type from a specific object. Xingyi Zhou [11] proposed a category-agnostic keypoint representation, which combines a multi-peak heatmap (StarMap) for all the types of keypoints in Pascal3D+ dataset [8] using the hourglass network [15]. This representation provides the flexibility to represent varying numbers of keypoints across different categories. Despite the strong generalization performance, this method cannot provide semantic information of the keypoints.

3. Method

The framework of the method proposed is shown in Figure 1, which consists of two parts: keypoint detection and deformable model matching processing.

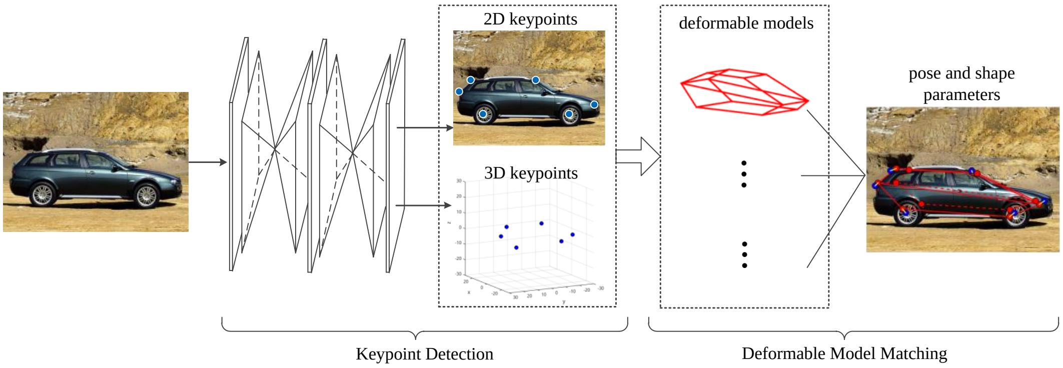

Figure 1. Illustration of the framework. For an input image, 2D keypoints and their 3D coordinates are obtained through the hourglass network. These keypoints are then matched with deformable models. After that, the viewpoint and shape parameters are obtained.

In the first part, the hourglass network [15] is used to predict keypoints from an input image with two components: 2D location and their 3D coordinates. Network training is illustrated in Section 3.1. In the second part, the keypoints detected by the network are matched with deformable models, then parameters of pose and shape are estimated. The formation of deformable model matching is shown in Section 3.2.1. Priori structures of multi-class objects are used in the matching processing, which are represented by the deformable wireframe models based on PCA. The building of deformable wireframe models is introduced in Section 3.2.2. Parameter estimation of pose and shape is introduced in Section 3.2.3.

The rest of this section describes the keypoint detection and the deformable model matching in detail.

3.1. Keypoint Detection Network

For keypoint detection, the most widely used way is to represent keypoints as multi-channel heatmaps, which associate each keypoint with one channel on a specific object category. For these methods, although each keypoint is semantically meaningful, they are limited in the specific category. In other words, keypoints from different objects are completely separated. We aim to detect keypoints across different categories, so a generalized network for multi-class objects can be obtained. This approach is inspired by the category-agnostic keypoint detection approach network proposed by Xingyi Zhou [11]. This method can locate all keypoints across different categories using only one network, but keypoints obtained have no semantic meaning. For 3D viewpoint estimation, the semantic meaning of each keypoint is needed to match with a priori model. As a result, keypoints, their 3D location, and depth are needed to obtain the semantic meaning of keypoints in Zhou's work. Network in our method is similar to Zhou, while we only need 2D keypoints and their 3D location during training. This is because the semantic meaning of each keypoint can be given by matching with a deformable model behind. The network used is shown in Figure 2.

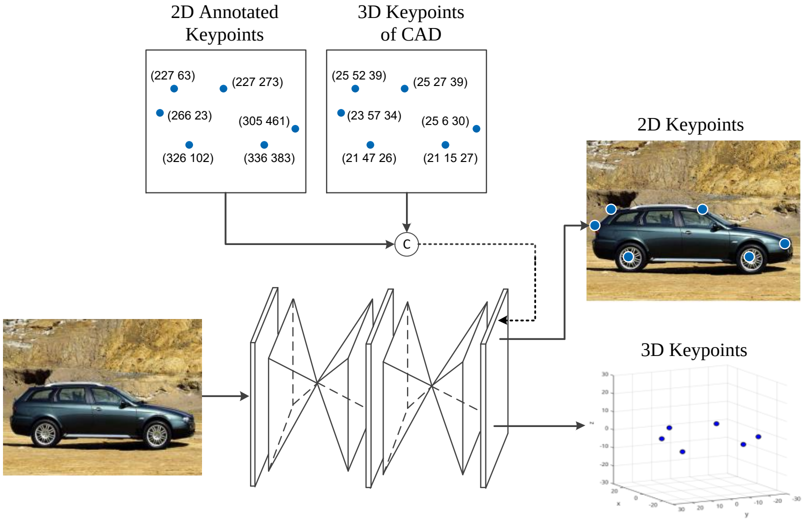

Figure 2. Hourglass network for keypoint detection. The network of our method predicts 2D keypoints and their 3D coordinates.

Training the network in our method requires annotations of 2D keypoints and their corresponding 3D locations. Annotations of 2D keypoints per image are widely available in many datasets. Annotating 3D keypoints of a CAD model is also not hard work with an interactive 3D UI, which has been done in some dataset such as Pascal3D+ and ObjectNet3D dataset [16]. Compared with Zhou's work, data preparation for the network is more feasible. A 2-stacks hourglass network is used. The 2D keypoints and their 3D locations are allocated by four-channel heatmaps. During training, the L2 distance is minimized between the output four-channel heatmaps and their ground truth.

3.2. Deformable Model Matching

After the keypoint detection network, the next step is to estimate object parameters from keypoints. The keypoints detected from the network have no semantic meaning. Thus, only the keypoints are not enough to estimate pose and shape parameters for multi-object. Besides, the following parameters are needed for viewpoint estimation and 3D modeling: object category, semantic meaning of keypoint, shape parameters, and 3D pose. It is difficult to obtain all the parameters simultaneously.

In order to solve these problems, we propose the deformable model matching method shown in Figure 3. Deformable model matching can be divided into two stages: model selection and model validation. In the first stage, the extracted 3D keypoints are utilized to match with different category of deformable models for the selection of candidate deformable models. In the second stage, these candidate models are validated by the matching with the extracted 2D keypoints, and shape and viewpoint parameters can be obtained by optimization. The final object parameters are obtained from the best fitting deformable model.

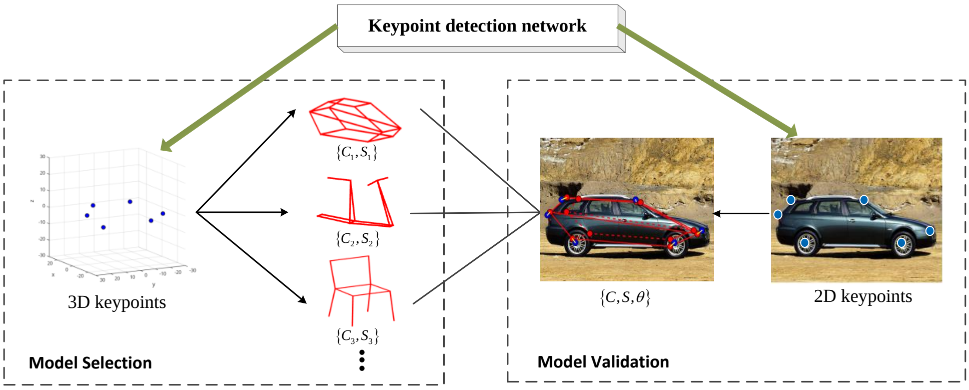

Figure 3. Deformable model matching processing. Deformable model matching can be divided into two stages: model selection and model validation. The object parameters are obtained from the best fitting deformable model.

The formulation, model building, and optimization of the deformable model are described in the following sections.

3.2.1. Formulation

The following demonstrates how to obtain object attribute parameters from keypoints including C, S, and R. C indicates the object category, S is shape parameter vector defined in Equation (7), and R indicates a rotation matrix of object pose. $P(R,S,C)$ indicates the probability distribution of object parameters. It is an optimal solution of target parameters by maximum the probability:

$$\{R,S,C\} = \max_{R,S,C} P(R,S,C) \tag{1}$$

The deformable matching processing consists of model selection and model validation, which correspond to the prior probability and the conditional probability respectively:

$$P(R,S,C) = P(C) \cdot P(R,S|C) \tag{2}$$

$P(C)$ is the probability of selecting the deformable model that is defined as

$$P(C) \propto e^{-\lambda_1 \cdot \min\limits_{S}(\sum\limits_{i} \|\mathbf{M}_i^C(S) - \mathbf{X}_i\|^2)} \tag{3}$$

where $\mathbf{M}$ indicates the node point coordinates of deformable wireframe model, which is defined in detail in Equations (6) and (7). $\mathbf{X}$ represents 3D keypoint coordinates obtained by the network. λ_1 is a constant. The probability of belonging to a certain class target is related to the distance between the 3D keypoints and deformable model point set.

$P(R,S|C)$ is the probability distribution of object parameters under the deformable model of a certain category, which is defined as

$$P(R,S|C) \propto e^{-\lambda_2 \cdot \sum_i \|\mathbf{m}_i(S)-\mathbf{x}_i\|^2} \tag{4}$$

where **m** indicates the projection points of **M** based on camera imaging model, and **x** represents 2D keypoint coordinates obtained by the network. λ_2 is a constant. Substituting Equations (2)–(4) into Equation (1), we have

$$\{R,S,C\} = \max_{R,S,C} e^{-\lambda_1 \cdot \min\limits_S (\sum_i \|\mathbf{M}_i^C(S)-\mathbf{X}_i\|^2)} \cdot e^{-\lambda_2 \cdot \sum_i \|\mathbf{m}_i(S)-\mathbf{x}_i\|^2} \tag{5}$$

It is worth mentioning that only the visible points are considered. Invisible points in **M** and **m** are discarded in Equations (3)–(5). In our method, the visibility of one node point in **M** or **m** is related to the distance with its nearest keypoint.

3.2.2. Model Building

In this section, the building of multi-class deformable models is illustrated. The deformable model is expected to have the ability to capture intraclass variance. We model each object category as a deformable 3D wireframe that is concise and expressive. During training, 3D keypoint coordinates annotated in a CAD model are represented as a vector, and we perform PCA on these vectors for CAD model library of a certain category. The geometry representation is based on the mean wireframe μ plus a linear combination of r principal components p_k with geometry parameters s, where s_k is the weight of the k th principal component:

$$\mathbf{M}(S) = \mu + \sum_{k=1}^{r} s_k p_k \tag{6}$$

The 3D wireframe can be determined by shape parameter S:

$$S = \{s_k\}_{k=1...r} \tag{7}$$

r is set as 3 in this paper. An example of a 3D wireframe model is shown in Figure 4.

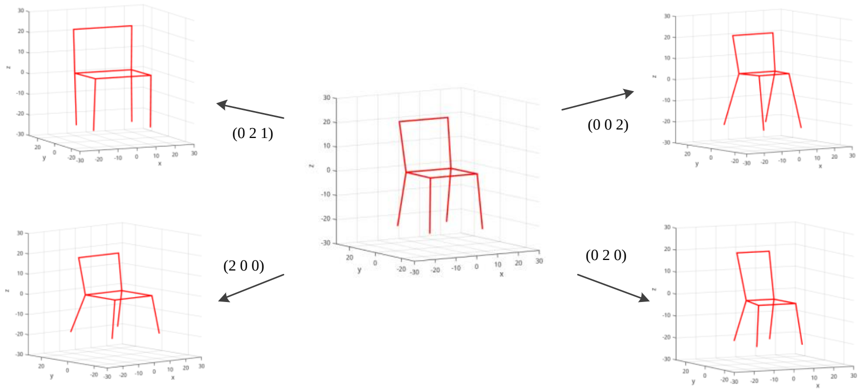

Figure 4. Deformable representation of a 3D wireframe. Chair models of different shape parameters are generated by PCA. Numbers indicated in the figure are weighting parameters of the first three principal component directions, which are represented as S.

3.2.3. Optimization

We use random hill climbing to solve the optimization problems in Equation (5). The probability distribution of Equation (5) consists of two relatively independent items. To make the results optimized and escape from the trap of local minimum, stepwise optimization is our strategy. For the first item, the probability of Equation (3) is obtained by matching 3D keypoints with each deformable model.

Qualitative results are shown in Figure 5. From the result, we can see that keypoints are fitted well with the deformable model through optimization.

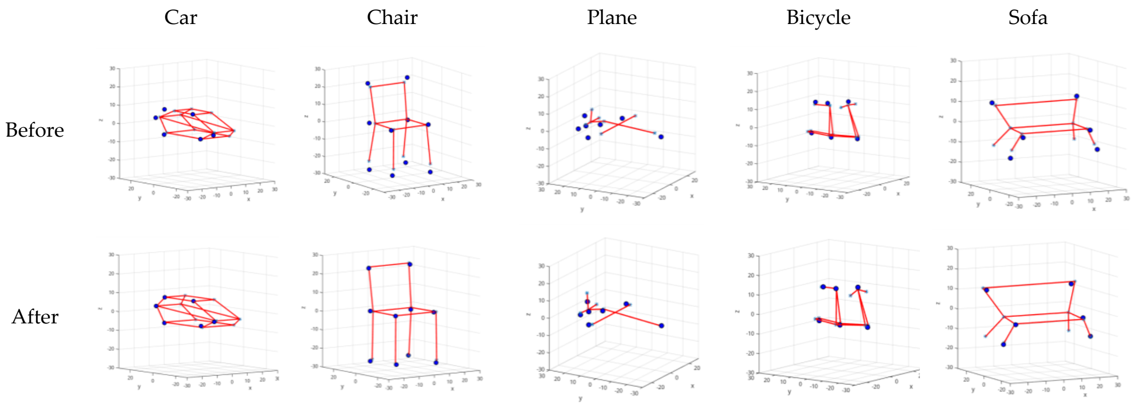

Figure 5. Qualitative results of matching 3D keypoints with the deformable model. It can be seen that 3D keypoints and corresponding deformable models are well fitted after matching.

The optimization of the second item in Equation (5) is sensitive to the initial value of viewpoint parameters. We propose a method for determining the initial value by visible keypoints. Taking a car as an example, Figure 6 illustrates the detailed procedure.

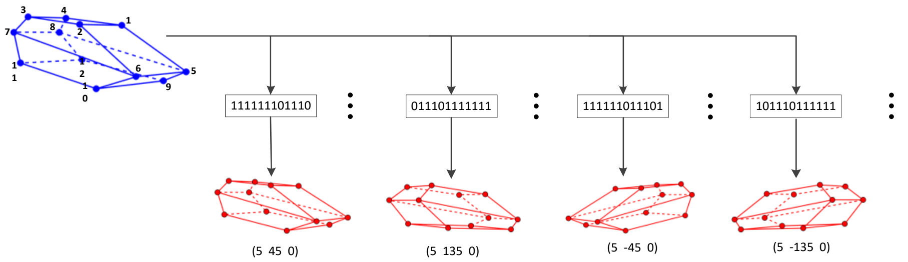

Figure 6. Initial viewpoint determination method. The visibility of keypoints in the different viewpoints is counted as a dictionary. For an input point set, the initial viewpoint can be obtained by matching their visibility with the dictionary.

A discrimination mechanism is necessary for the objects that have different category than the priori models. For the objects whose category is out of our priori models, the deviation of deformable model projection and 2D keypoints would be larger compared with objects that have the same category with the corresponding deformable model. Consequently, a threshold is set for the deviation to judge whether it belongs to categories contained.

A summary of the method is shown in Figure 7.

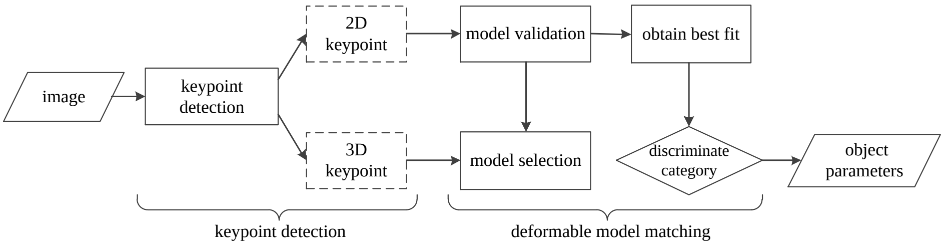

Figure 7. The summary of the method. At first, keypoints are detected from the image. The 3D keypoints are then matched with each deformable model and 2D keypoints are matched with the projection of the deformable model. After the matching process, the optimal parameters of pose and shape are obtained from the best fit deformable model. Next step is to judge whether the target type belongs to the existing models according to the matching deviation. Finally, the object parameters are obtained.

4. Experiments

We evaluate the method proposed on Pascal3D+ dataset. In this section, we evaluate the approach from two aspects: wireframe modeling and viewpoint estimation. It is important to note that these two tasks are completed at the same time in our method. Our implementation is done in the PyTorch framework and Matlab2014.

4.1. *Wireframe Modeling*

The qualitative results of the wireframe modeling are shown in Figure 8.

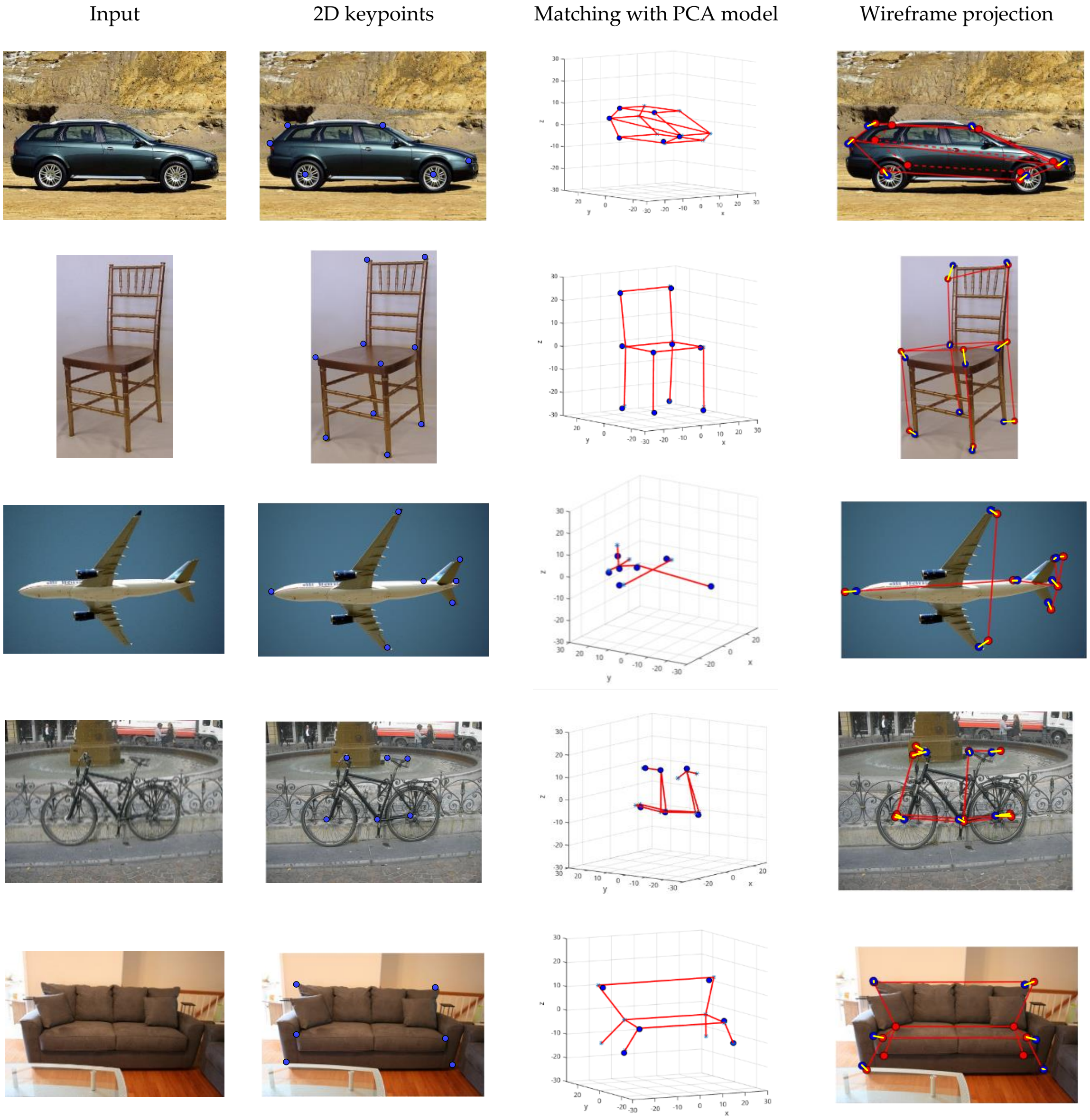

Figure 8. *Cont.*

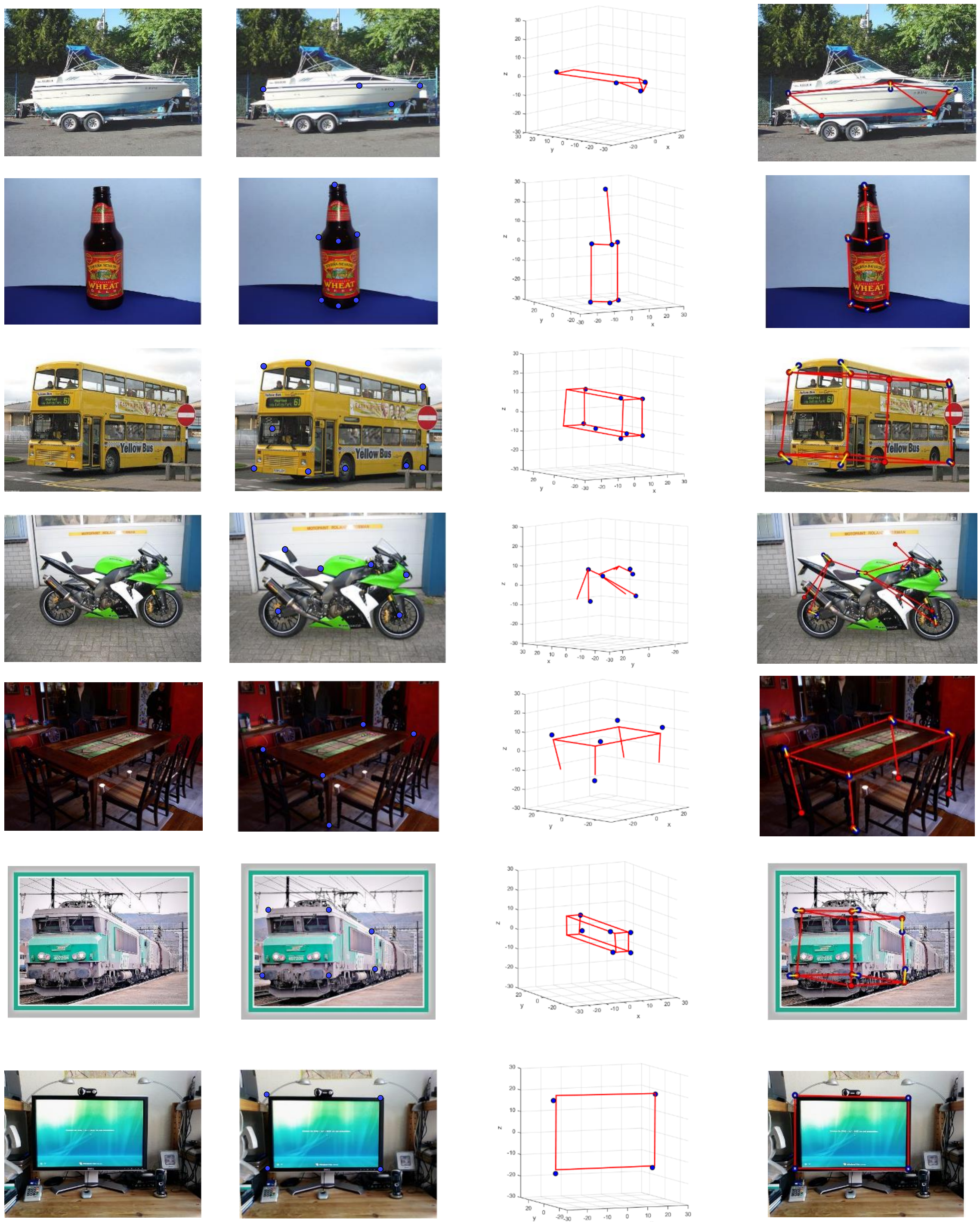

Figure 8. Qualitative results of 3D wireframe modeling. Four columns in the figure are input image, 2D keypoints detected, matching between 3D keypoints and deformable model, and wireframe projection. In the fourth column, the red line represents the projection of wireframe, and its deviation with 2D keypoints detected is shown as the yellow line.

The 3D wireframe models are generated after the deformable model matching. In order to test the robustness, we artificially insert the error results in keypoints obtained by the network. The deviation between the wireframe model projection and the ground truth of 2D keypoints is evaluated. Take the case of a sofa, the results of which are shown in Figure 9.

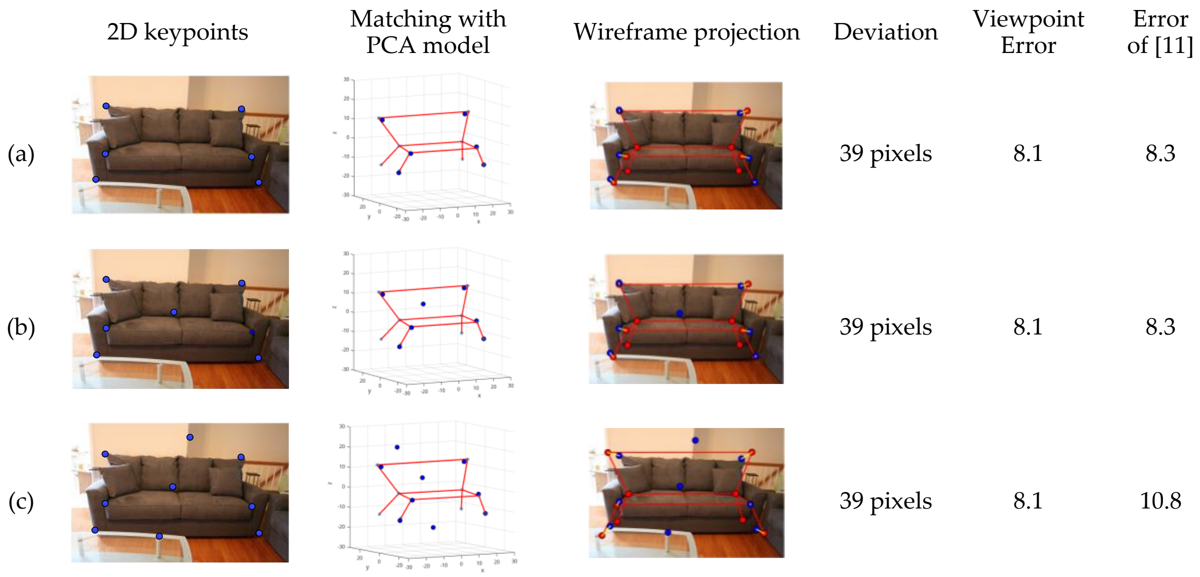

Figure 9. Fault tolerance test. We artificially added error keypoints to test the fault tolerance, as shown in the first and second column. There is no error keypoint in (**a**). (**b**) and (**c**) show cases with error keypoints added. Through the corresponding process presented in the second column, the method proposed can identify the mistaken points shown as the third column. For the method proposed, the deviation between wireframe model projection and ground truth of 2D keypoints does not change with the increase of mistaken points as displayed in the fourth column. The last two columns compare the fault tolerance of method in [11] and the method proposed on viewpoint estimation, which using evaluation criteria in Equation (8).

It can be seen that the method proposed can tolerate incorrect keypoints well. Because of the deformable model matching, the error results of the network do not seem to cause significant negative effects.

4.2. Viewpoint Estimation

For viewpoint estimation, the angle error between the predicted rotation vector and the ground truth rotation vector is measured as

$$\Delta\left(R_{pred}, R_{GT}\right) = \frac{\|\log(R_{pred}^{T} R_{GT})\|_F}{\sqrt{2}} \tag{8}$$

R is the rotation matrix along X, Y, and Z axis. We consider Median Error and Accuracy as two evaluation criteria that are commonly applied in the literature. Median Error is the median of the rotation angle error, and Accuracy at θ is the percentage of objects whose error is less than θ. θ is set as $\pi/6$ in this paper.

Any image of multi-class targets can be processed by the method proposed with only one network with the deformable model matching processing. For comparison purposes, the results of each category are counted in Table 1.

Table 1. Results of viewpoint estimation.

		Car	Chair	Aero	Bike	Sofa	Boat	Bottle	Bus	Table	Train	Tv	Motor	Mean
Median Error	Tulsiani [17]	9.1	14.8	13.8	17.7	13.7	21.3	12.9	5.8	15.2	8.7	15.4	14.7	13.6
	Mousavian [18]	5.8	11.9	13.6	12.5	12.8	22.8	8.3	3.1	12.5	6.3	11.9	12.3	11.1
	Zhou X	6.5	11.0	10.1	14.5	11.1	30.0	9.1	3.1	23.7	7.4	13.0	14.1	10.4
	Method proposed	6.3	10.8	10.4	14.6	10.9	23	8.9	3.2	14	7.1	12.2	13.9	10
Accuracy	Tulsiani [17]	0.89	0.80	0.81	0.77	0.82	0.59	0.93	0.98	0.62	0.80	0.80	0.88	0.806
	Mousavian [18]	0.90	0.80	0.78	0.83	0.82	0.57	0.93	0.94	0.68	0.82	0.85	0.86	0.810
	Zhou X	0.92	0.79	0.82	0.86	0.92	0.50	0.92	0.97	0.62	0.77	0.83	0.88	0.823
	Ours	0.93	0.81	0.81	0.8	0.92	0.52	0.93	0.97	0.63	0.78	0.84	0.89	0.829

From Table 1 it can be seen that the method proposed performs as well as mainstream approaches in viewpoint estimation. It should be noted that the first two methods are class-specific, while ours and Zhou's methods are designed for multi-class objects and have better universality.

From the result, the accuracy of the method proposed is similar to Zhou, but we take a completely different solution. The 3D keypoints from the network are used to estimate viewpoint directly in [11], while 3D keypoints are only used to obtain semantic meaning and shape initial value. Viewpoint estimation is conducted in deformable model matching in our method. As a result, the method in [11] relies heavily on 3D keypoints and is vulnerable to mistaken points, while the method proposed here provides better fault tolerance ability. This conclusion is verified by the following two experiments.

Firstly, the ability to identify mistaken keypoints is compared in Figure 10. It can be seen that the method proposed can distinguish mistaken keypoints by the optimization process, while it is difficult for the method in [11].

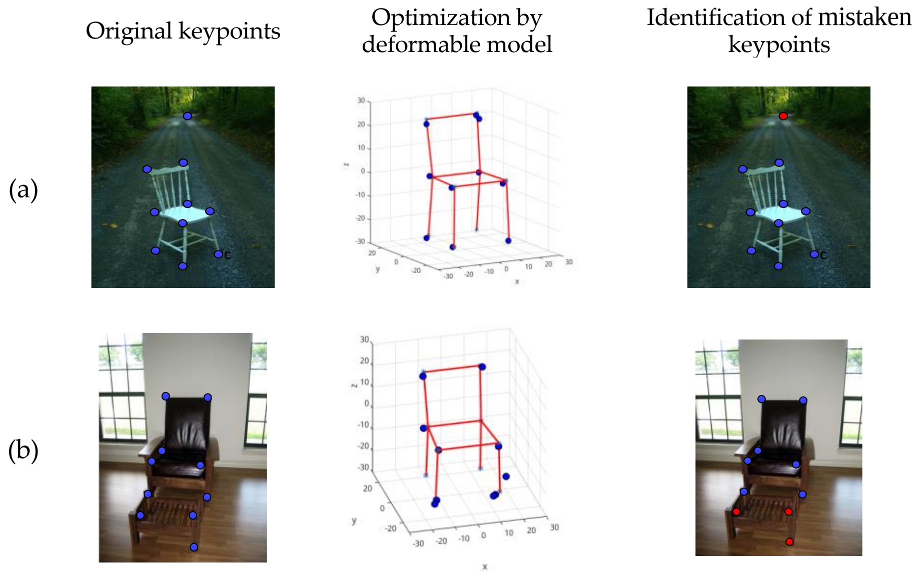
Original keypoints
Optimization by deformable model
Identification of mistaken keypoints
(a)
(b)
x
y
z

Figure 10. Identification of mistaken keypoints. (**a**) and (**b**) are two image cases. Keypoints from the network are displayed as the first column, the mistaken points cannot be detected in [11]. The method proposed can identify mistaken keypoints though optimization process in the second column. Mistaken keypoints are shown as red points in the last column.

Next, we evaluate the fault tolerance ability of viewpoint estimation. As shown in the last two columns in Figure 9, viewpoint estimation of the method in [11] is sensitive to mistaken keypoints, while our method provides better fault tolerance. More quantitative results are shown in Figure 11.

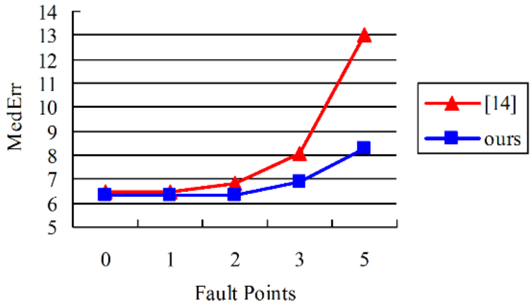

Figure 11. Comparison of viewpoint estimation with error keypoints using data from Car in Pascal3D+ dataset. We artificially add error keypoints to test the method's fault tolerance.

It can be seen that as the number of error keypoints increases, the accuracy of [11] decreases faster than the method proposed. This is because 3D keypoints are used to estimate the viewpoint directly in [11], while our method conducts viewpoint estimation by deformable model matching, which is more robust.

Finally, we test the performance of the method proposed for occluded object image cases. In practice, it is impossible for objects to be always visible. To evaluate our method, we artificially occlude the image of the car in Pascal3D+ dataset. The result is shown in Figure 12.

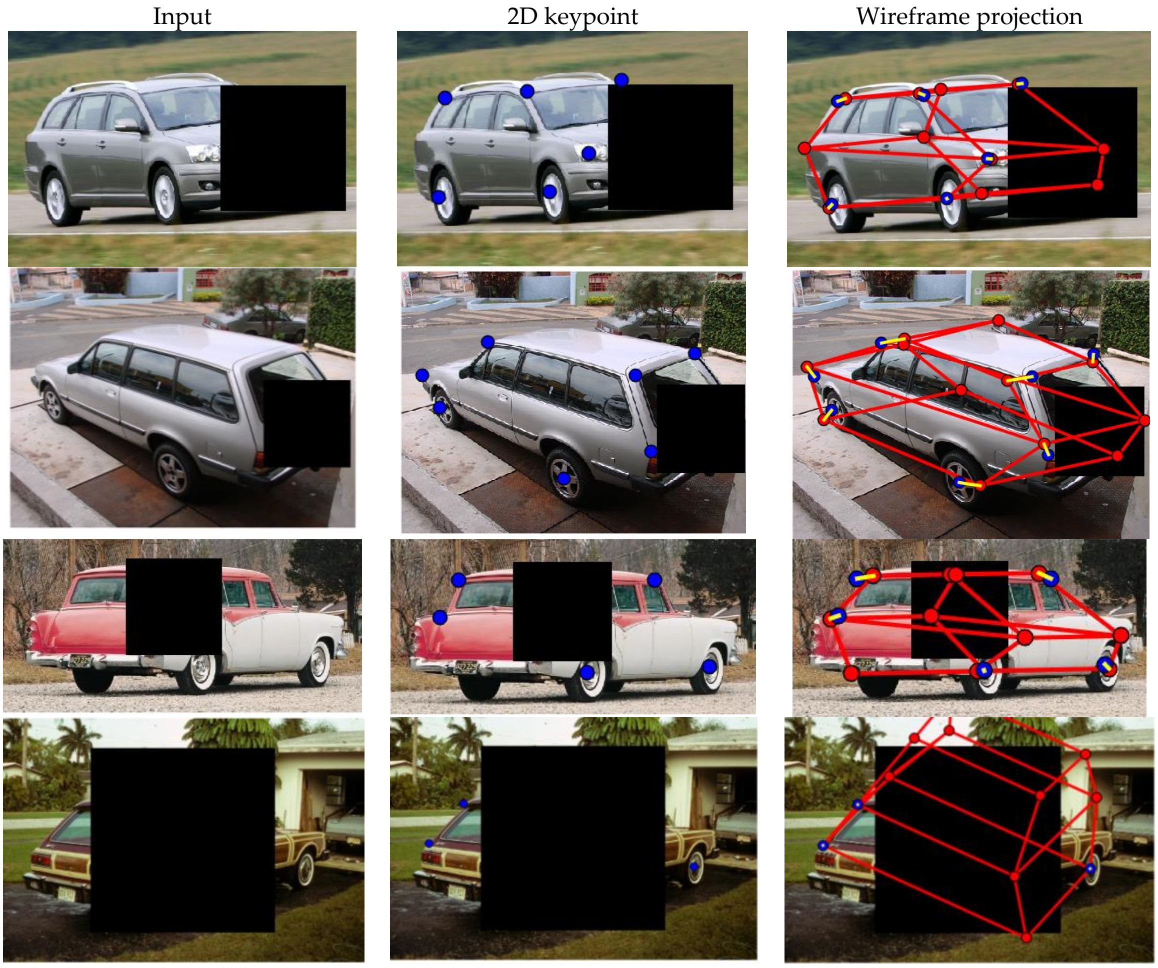

Figure 12. Qualitative results for occluded object image cases. Three columns in the figure are input image, 2D keypoints detected and wireframe projection. The last instance is a failure case.

The median error is 9.5, which is larger than cases without occlusion. It can be seen from the results that the method proposed can work with some keypoints missing caused by occlusion, although we have some failure cases for large occlusion. Because of the priori deformable model, our approach can tolerate the absence of some keypoints to a certain extent.

4.3. Computation

Table 2 presents the elapsed statistics of the method proposed.

Table 2. The elapsed time.

Class Number	2	4	6	8	10	12
Time (s)	1.1	1.3	1.5	1.7	1.9	2

To evaluate the effects of the category number, we test the time consumption in different object category number. The results show that with an increase of the number of categories, time consumption rises because the computation of point matching increases.

4.4. Discussion

The experimental results show that the method proposed has the following advantages. Firstly, our method has a strong generalization ability. Compared with category-specific methods, there is only one CNN with deformable model matching processing for the 3D viewpoint and the shape estimation for all the types of objects in Pascal3D+ dataset. Secondly, the method proposed has a robust fault-tolerant ability. Similar to many methods, we estimate the 3D viewpoint depending on the detection of keypoints. Compared with methods such as [11], our method has better fault-tolerance to mistaken keypoints, as shown in Section 4.2. This is a result of the priori object structure and optimization mechanism in deformable model matching. Mistaken keypoints from the network can be eliminated after the matching with deformable models.

5. Conclusions

In this paper, a 3D viewpoint and shape estimation method for multi-class objects is proposed. The method proposed combines the advantages of the data-based method and model-based method and conducts wireframe modeling and viewpoint estimation through maximizing probability distribution. Compared with the methods limited in a specific category, our method only uses one keypoint detection network with the deformable model matching processing for multi-class objects. Experiments on Pascal3D+ dataset show that the method proposed performs well in accuracy and generalization. Besides, due to the deformable model matching processing, the method proposed has robust fault-tolerance to mistaken keypoints detected from the network. Our research is valuable in exploration to extend the generalization of deep learning in specific tasks.

Author Contributions: Conceptualization, X.R.; methodology, X.R.; software, L.J.; validation, W.L.; formal analysis, W.L.; investigation, X.R.; resources, L.J.; data curation, X.T.; writing—original draft preparation, X.R.; writing—review and editing, L.J.; visualization, L.J.; supervision, X.T.; project administration, X.T.; funding acquisition, X.T.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Zhou, X.; Zhu, M.; Leonardos, S.; Daniilidis, K. Sparse representation for 3D shape estimation: A convex relaxation approach. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1648–1661. [CrossRef] [PubMed]

2. Roberts, L.G. Machine Perception of Three-Dimensional Solids. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1963.
3. Brooks, R.A. Symbolic reasoning among 3-D models and 2-D images. *Artif. Intell.* **1981**, *17*, 285–348. [CrossRef]
4. Zhang, X.; Jiang, Z.; Zhang, H.; Wei, Q. Vision-based pose estimation for textureless space objects by contour points matching. *IEEE Trans. Aerosp. Electron. Syst.* **2018**, *54*, 2342–2355. [CrossRef]
5. Zhang, H.; Jiang, Z.; Yao, Y.; Meng, G. Vision-based pose estimation for space objects by Gaussian process regression. In Proceedings of the IEEE Aerospace Conference, Big Sky, MT, USA, 7–14 March 2015; pp. 1–9.
6. Cao, Z.; Sheikh, Y.; Banerjee, N.K. Real-time scalable 6DOF pose estimation for textureless objects. In Proceedings of the International Conference on Robotics and Automation, Stockholm, Sweden, 16–21 May 2016; pp. 2441–2448.
7. Pepik, B.; Stark, M.; Gehler, P.; Schiele, B. Teaching 3D geometry to deformable part models. In Proceedings of the Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3362–3369.
8. Xiang, Y.; Mottaghi, R.; Savarese, S. Beyond PASCAL: A benchmark for 3D object detection in the wild. In Proceedings of the Workshop on Applications of Computer Vision, Steamboat Springs, CO, USA, 24–26 March 2014; pp. 75–82.
9. Wu, J.; Xue, T.; Lim, J.J.; Tian, Y.; Tenenbaum, J.B.; Torralba, A.; Freeman, W.T. 3D Interpreter networks for viewer-centered wireframe modeling. *Int. J. Comput. Vis.* **2018**, *126*, 1009–1026. [CrossRef]
10. Li, C.; Zeeshan Zia, M.; Tran, Q.H.; Yu, X.; Hager, G.D.; Chandraker, M. Deep supervision with shape concepts for occlusion-aware 3D object parsing. In Proceedings of the Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 388–397.
11. Zhou, X.; Karpur, A.; Luo, L.; Huang, Q. StarMap for category-agnostic keypoint and viewpoint Estimation. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 328–345.
12. Felzenszwalb, P.F.; Girshick, R.B.; McAllester, D.; Ramanan, D. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 1627–1645. [CrossRef] [PubMed]
13. Toshev, A.; Szegedy, C. DeepPose: Human pose estimation via deep Neural networks. In Proceedings of the Computer Vision and Pattern Recognition, Honolulu, HI, USA, 24–27 June 2014; pp. 1653–1660.
14. Yu, X.; Zhou, F.; Chandraker, M. Deep deformation network for object landmark localization. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 52–70.
15. Newell, A.; Yang, K.; Deng, J. Stacked hourglass networks for human pose estimation. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 483–499.
16. Xiang, Y.; Kim, W.; Chen, W.; Ji, J.; Choy, C.; Su, H.; Savarese, S. ObjectNet3D: A large-scale database for 3D object recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October; pp. 160–176.
17. Tulsiani, S.; Malik, J. Viewpoints and keypoints. In Proceedings of the Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1510–1519.
18. Mousavian, A.; Anguelov, D.; Flynn, J.; Kosecka, J. 3d bounding box estimation using deep learning and geometry. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 5632–5640.