Improving a Deep Learning based RGB-D Object Recognition Model by Ensemble Learning

Aakerberg, Andreas; Nasrollahi, Kamal; Heder, Thomas

Published in:
2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)

DOI (link to publication from Publisher):
10.1109/IPTA.2017.8310101

Publication date:
2018

Document Version
Accepted author manuscript, peer reviewed version

Improving a Deep Learning based RGB-D Object Recognition Model by Ensemble Learning

Andreas Aakerberg[1], Kamal Nasrollahi[2] and Thomas Heder[1]
[1] HSA Systems, Aalborg, Denmark
e-mail: aakerberg@me.com, th@hsasystems.com
[2] Visual Analysis of People Laboratory, Aalborg University, Denmark
e-mail: kn@create.aau.dk

***Abstract*— Augmenting RGB images with depth information is a well-known method to significantly improve the recognition accuracy of object recognition models. Another method to improve the performance of visual recognition models is ensemble learning. However, this method has not been widely explored in combination with deep convolutional neural network based RGB-D object recognition models. Hence, in this paper, we form different ensembles of complementary deep convolutional neural network models, and show that this can be used to increase the recognition performance beyond existing limits. Experiments on the Washington RGB-D Object Dataset show that our best performing ensemble improves the recognition performance with 0.7% compared to using the baseline model alone.**

***Keywords*— Deep Learning, Computer Vision, RGB-D, Convolutional Neural Networks, Ensemble Learning.**

I. Introduction

In this paper, we address the problem of RGB-D based object recognition, which deals with making a machine capable of identifying object types using both RGB and depth data. Although successful RGB based object recognition models already exist, recent advancements within range imaging technologies have made supplemental depth data available, which can be used to further increase the recognition performance. This is possible, as the depth data contains additional geometric information about the object shapes, besides the texture, color and appearance information already contained in the RGB data. The depth data is furthermore invariant to lighting and color variations, allowing for a potentially more robust classifier [Guo et al., 2014]. Current State-of-the-Art (SoTA) methods within both RGB and RGB-D object recognition mainly relies on deep Convolutional Neural Network (CNN) as feature extractors, as these are generally superior to the classical methods such as Scale Invariant Feature Transform (SIFT) [Lowe, 2004] and Histogram of Oriented Gradients (HOG) [Dalal and Triggs, 2005] [Razavian et al., 2014]. RGB-based object recognition models are typically evaluated on the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) dataset [Russakovsky et al., 2015], and often make use of ensemble learning to effectively minimize the prediction error. In fact, all the latest winning entries on the ILSVRC are ensembles of CNN models. A widely used dataset for evaluation of RGB-D based object recognition models is the Washington RGB-D Object Dataset [Lai et al., 2011]. However, literature experimenting with improving the performance of RGB-D based object recognition models using ensemble learning is sparse. Surprisingly, to the best of the author's knowledge, there currently does not exist any work describing the effect of using ensemble learning in combination with CNN based RGB-D object recognition models in the literature. Hence, we have conducted experiments under the hypothesis that ensemble learning can also be beneficial within the RGB-D domain. We find that the recognition performance of an existing multi-modal RGB-D object recognition model can be increased significantly by forming an ensemble of two generalist models and an expert model, when evaluated on the publicly known Washington RGB-D Object Dataset.

II. Related Work

Our method is related to work within the field of both RGB and RGB-D based object recognition, and ensemble learning.

Ensemble Learning Theory. Ensemble learning is a way to build a stronger model, by combining a collection of weaker and diverse models to get an aggregated prediction, a concept commonly used with the highly successful deep CNNs, as these models generally have a high variance and a low bias [Dietterich, 2000]. By averaging the prediction of slightly uncorrelated variants of these models, the variance can be reduced significantly. Namely, the diversity between the models is crucial in order to improve the performance by averaging the predictions [Bishop, 2006].

Ensembles of CNNs. Ensemble learning is widely used by the winning entries in the prestigious image classification challenge ILSVRC. In [Krizhevsky et al., 2012], an ensemble of 5 identical but differently trained versions of the AlexNet, achieved a top-1 error rate of 38.1%, compared to a single CNN with an error rate of 40.7% on the 2012 ILSVRC. The same tendency was also found in [Simonyan and Zisserman, 2014], where VGGNet models trained with different initialization was used to form an ensemble. In [Szegedy et al., 2014], an ensemble of 7 GoogLeNet model, trained with a differently sampled dataset resulted in a 3.45% reduction of the top-5 error on the 2014 ILSVRC. In [He et al., 2015] an ensemble of six ResNet models with different depths reduced the top-5 error with 0.92% on the 2015 ILSVRC. However, none of these ensemble approaches actively tries to combine individual models which best complement each other, but instead relies on the sheer

amount of individual models in the ensembles. According to [Bonab and Can, 2016] the number of models in an ensemble should be the same as the amount of classes, to obtain the highest possible accuracy. However, in many cases this is impractical, and for evaluation on the Washington RGB-D dataset this would mean that 510 individual models would have to be trained, as this particular dataset contains 51 different classes and uses 10-split cross-validation.

RGB-B Object Recognition. One of the first uses of CNNs for RGB-D image classification was the work of [Socher et al., 2012], where a model based on a CNN combined with a Recursive Neural Network (RNN) was used as feature extractors in combination with a Support Vector Machine (SVM) classifier. In [Eitel et al., 2015] an end-to-end mapping from image pixels to object classes is performed using a two-stream CNN, operating on RGB and depth data respectively, and an additional late fusion network and classifier. A simple, but effective Jet encoding of the depth values enabled the use of models pre-trained on ImageNet data. In [Li et al., 2015] dense local features are extracted from the depth data and encoded as Fisher vectors. These features are concatenated with RGB features extracted by a CNN, and fed to a SVM classifier. In [Carlucci et al., 2016] a large database, with more than 4 million synthesized depth images, is created for the purpose of training a CNN on the raw depth data, without the need of prior pre-processing. A method that resembles the one of [Eitel et al., 2015], but relies on deeper networks and surface normal encoding of the depth values are presented in [Wang et al., 2016]. In [Sun et al., 2017] a model pre-trained on virtual Computer-aided design (CAD) data used to eliminate the need for color encoding of the depth data. In [Asif et al., 2017] hierarchical cascaded forests are used both to compute grasp-poses and predict object categories.

III. Proposed Approach

Ensembles of CNNs are typically created by averaging a relatively high number of independent models, which are often created by slightly changing the models hyper-parameters or the order of which the training samples are presented. However, creating a sufficient amount of independent models this way, for evaluation on the Washington RGB-D Object Dataset is not feasible and also inefficient. To this end, we form our ensembles using candidate models which are known in advance, to be complementary each other. This enables us to use much fewer models in the ensemble, while still providing an improvement in recognition accuracy [Lee et al., 2015]. To establish the knowledge to accomplish selecting proper ensemble candidates, we first study the baseline model and different pre-processing methods of the depth-images. Finally, we review commonly used ensemble methods.

A. Baseline Model

The baseline model for our work is the deep learning based multi-modal object recognition model described in [Aakerberg et al., 2017]. This model is based on the FusionNet concept, proposed by [Eitel et al., 2015], consisting of two CNNs streams, pre-trained on ImageNet data, operating on RGB and depth data respectively. A late fusion approach is used to combine features extracted by the two streams, effectively creating a multi-modal classifier which creates higher level representations of features from the two modalities. Different from [Eitel et al., 2015] this model uses a deeper network architecture for the RGB stream, namely the 16-layered VGGNet [Simonyan and Zisserman, 2014] in comparison to the 8-layered CaffeNet [Jeff Donahue,] used in the original FusionNet, and rely on colorized surface normals for encoding of the depth values. When evaluated on the Washington RGB-D Object Dataset this model has a recognition performance of 93.5 ± 1.1. Furthermore, as seen in Fig. 1, this model has a recall that is $>98\%$ on 31 out of the 51 classes in the dataset.

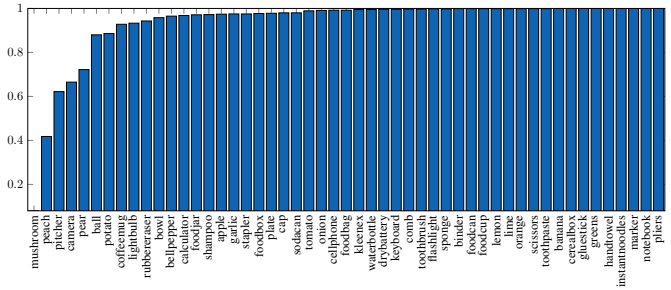
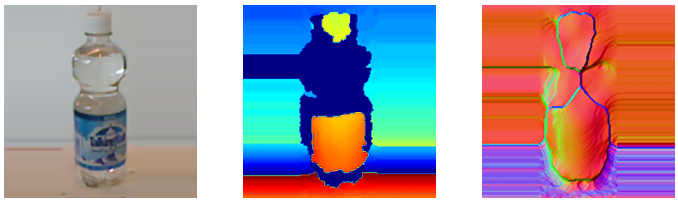


Fig. 1: The per-class recall of the baseline model, averaged over all ten splits.

B. RGB-D Image Pre-Processing

Both the RGB and depth data needs to be pre-processed before it can be used in combination with CNNs pre-trained on ImageNet data. The baseline model used in this work does this by squaring images from both domains using border replication of pixels on the longer sides and re-sizing them to 256×256 pixels. During training and inference, the images are either randomly cropped, or center cropped to match the input dimensions of the respective CNNs. While the RGB images need no further processing, the depth images have to be transferred to the RGB domain to benefit from the features learned in the CNNs pre-trained on natural images. The baseline model uses colorized surface normals to effectively capture object shape and curvature information. A drawback of this method is that the surface normals cannot be calculated correctly when large amounts of the depth values are missing, which is especially pronounced in depth images of objects with dark or highly specular surfaces. A simpler, but still effective, way to colorize the depth values is the Jet color encoding method proposed by [Eitel et al., 2015]. In comparison, this method, are able to preserve the outline of objects when large amounts of the depth values are missing. The two depth image pre-processing methods are visualized in Fig. 2.



RGB. Jet. Surface Normals.

Fig. 2: Comparison of Jet and surface normal encoding of the depth values, of an image with large amounts of missing depth values.

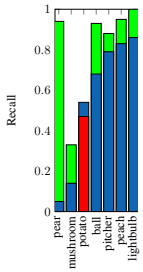


Fig. 3: Visualization of the improvement of the per-class recall on difficult classes from split-1, when training an expert model using only training images from these classes, in comparison to the baseline model. Green bars show the improvement in comparison to the baseline model, and red bars show classes where the expert model performs worse than the baseline model.

C. Ensemble Candidates

In this work, we propose three candidate models for forming ensembles, namely α, β and γ. α is the baseline model and β is a variant of the baseline model which uses Jet color encoding of the depth values, computed as described in [Eitel et al., 2015]. This latter has been trained similarly to the baseline model and has a recognition performance of $92.9\% \pm 1.0$ when evaluated on the Washington RGB-D Object Dataset. This is slightly lower than the baseline model, which is expected due to the performance differences between using surface normal or Jet-encoding of the depth values. Despite of this, β is still useful in an ensemble with the baseline model, as the two models have learned different but complementary features from the depth domain, and the fact that the Jet-encoding method tends to perform better with missing depth values. Additionally, we use the concept of generalist and expert models [Hinton et al., 2015], for the γ candidate, which is an expert model specialized in the particularly difficult classes of the dataset. This expert model is created by performing additional fine-tuning of the baseline model, but while only presenting training samples of classes with a recall lower than 94%. We fine-tune the expert model for 15,000 iterations with a learning rate of 0.001 which is dropped with a polynomial decay of the order 0.5, in conjunction with a momentum of 0.9 and a weight decay of 0.0005. Despite the fact that the expert models performance on the particular difficult classes is improved significantly, compared to the baseline model as seen in Fig. 3, the overall recognition performance of the expert model is considerably lower than the baseline model, due to the problem of catastrophic forgetting [Goodfellow et al., 2013]. However, by only including predictions from within the expert models domain, when performing inference with the ensemble, this problem can be mitigated.

D. Ensemble Methods

There exists a number of different ensemble methods where unweighted and weighted averaging and majority voting are among the most common ones.

Unweighted Averaging: The standard ensemble approach for CNNs is the unweighted average, where the softmax probabilities of each model are averaged to create the final prediction as seen in Equation 1. If the individual models in the ensemble are uncorrelated enough, the variance of the models will be reduced when averaging with a resulting increase in recognition performance.

$$P = \frac{1}{n}\sum_{i=1}^{n} softmax(i) \tag{1}$$

where $softmax(i)$ is the softmax score vector of the i-th model.

Weighted Averaging: The weighted averaging method resembles the unweighted averaging methods, with the one difference that the predictions from the individual candidates are weighted as seen in Equation 2. The weighted ensemble scheme is illustrated in Fig. 4.

$$P = \frac{1}{n}\sum_{i=1}^{n} \alpha_i softmax(i) \tag{2}$$

where $softmax(i)$ is the softmax score vector, and α_i the weight of the i-th model respectively.

The weight α_i can be determined in several different ways, including a grid search over all possible values. In this work, α_i is determined based on the individual candidate models performance on the validation set. Hence models with a high accuracy will have a large weight when averaging the softmax probabilities. To have weights which sum to 1 the weighted mean of the accuracy is used, which is calculated as seen in Equation 3.

$$\alpha_i = \frac{A_i}{\sum_{j=1}^{k} A_j} \tag{3}$$

where A_i is the accuracy on the validation set for model i, and k is the number of models in the ensemble.

Majority Voting: Majority voting can be used when the number of models in an ensemble is > 2. The aggregated prediction is created by counting the votes of all the predicted labels from the individual models and picking the one with the highest number of votes. This method is less sensitive to predictions from single models than the unweighted averaging method. In practice, majority voting is implemented by taking the mode of the top-1 predictions of all models. If the mode does not exist, i.e. all models predicted something different, the prediction of the strongest model is used.

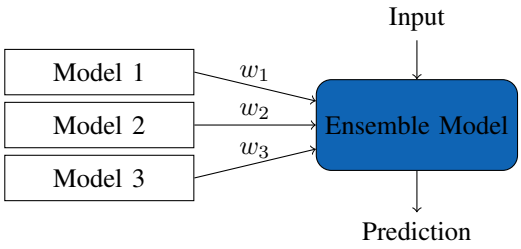


Fig. 4: Illustration of the weighted ensemble approach, where the weights w_1,w_2 and w_3 are used to weight each models contribution in the aggregated prediction.

IV. Experimental Results

We perform all our experiments using the Caffe deep learning framework [Jia et al., 2014], and use random cropping and horizontal flipping of the training images for data augmentation. During training and inference, we subtract the mean RGB and depth image from the input images, to center the data.

A. RGB-D Object Dataset

We use the Washington RGB-D object dataset [Lai et al., 2011] for training and evaluation of the proposed models and ensembles. This dataset contains 207,920 RGB-D images of common household objects, all captured in a controlled environment using a spinning table and a PrimeSense prototype RGB-D camera, similar to the Microsoft Kinect V1 camera. The RGB and depth information are stored in separate files, where the depth images files contain the depth in millimeters, stored in a single-channel image in the uint16 format, and the RGB information is stored in three-channel uint8 RGB images. The images are recorded continuously at 20 Hz and organized into 51 classes, which contains images of three to 10 different instances of objects of the same class, making a total of 300 distinct objects. There are several hundred images of each instance captured under three different viewpoint angles, namely 30°, 45°. and 60°. In combination with the dataset, the authors also present a method for subsampling the dataset, and 10 pre-defined training and test splits for cross-validation, which is adopted in this work, and nearly all SoTA works using this dataset. The dataset is subsampled by taking every fifth frame, resulting in 41,877 RGB-D images for training and evaluation. For each split, one random object instance from each class is left out from the training set and used for testing. Training is performed on images of the remaining $(300 - 51)$ 249 instances. This results in roughly 35,000 training images and 7,000 testing images in each split. At test time, the classifier has to assign the correct label to a previously unseen object instance from each of the 51 classes.

B. Ensembles

We form two ensembles, A and B, out of the three candidate models, using a weighted average of the individual model's softmax probabilities. Experiments with unweighted averaging have also been performed, but this resulted in $\approx 0.1\%$ lower recognition performance for both ensembles. Ensemble A consists of the α and β candidates, which are combined using the weights 0.57 and 0.43 respectively, found empirically using a validation set. Ensemble B consists of all three ensemble candidates. γ is however only included in the aggregated prediction for classes within its field of expertise. Here, we use the weights 0.17, 0.13 and 0.7 for $alpha$, $beta$ and $gamma$ respectively. The best increase in the recognition performance is achieved when using ensemble B, which results in an accuracy of $94.2\%\pm0.7$, 0.7% higher compared to the baseline model alone. The performance of ensemble A is slightly lower than ensemble B, namely with an accuracy of $93.7\% \pm 1.1$. Table 1 shows the performance of all the proposed RGB-D object recognition models in comparison to SoTA works.

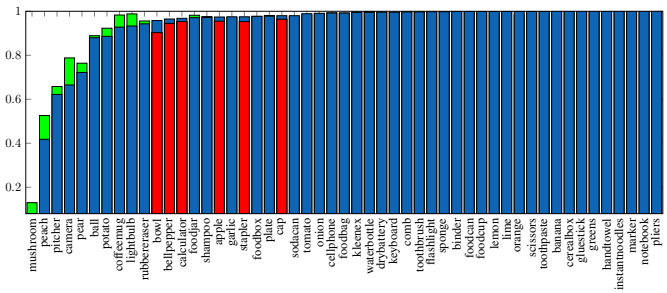


Fig. 5: Visualization of the improvement of the per-class recall made on average over all ten splits, when using ensemble B compared to the baseline model. Green bars show the improvement in recall, and red bars show classes where the ensemble performs worse than the baseline model.

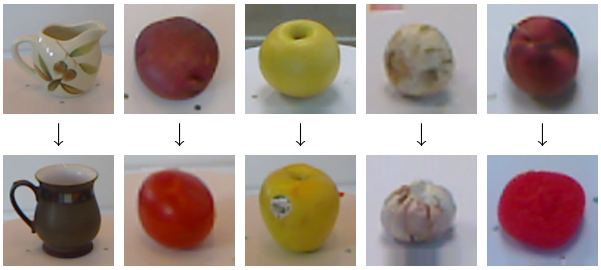
V. Discussion and Future Work

While the use of ensemble learning improves the recognition performance, there are still classes within the dataset which the proposed models often misclassifies. Examples of class instances which are often confused can be seen in Fig. 6. In this work, only a single expert model has been used in combination with generalist models to improve the recognition accuracy. One could train and include additional expert models, each trained to be experts in their own part of the

Table 1: Comparison of the recognition performance of the baseline and ensemble models proposed in this work to SoTA works. Red and blue indicates best and second best performance respectively.

Method	**RGB**	**Depth**	**RGB-D**
Nonlinear SVM [Lai et al., 2011]	74.5 ± 3.1	64.7 ± 2.2	83.9 ± 3.5
CNN-RNN [Socher et al., 2012]	80.8 ± 4.2	78.9 ± 3.8	86.8 ± 3.3
FusionNet [Eitel et al., 2015]	84.1 ± 2.7	83.8 ± 2.7	91.3 ± 1.4
CNN+Fisher [Li et al., 2015]	**90.8 ± 1.6**	81.8 ± 2.4	**93.8 ± 0.9**
DepthNet [Carlucci et al., 2016]	88.4 ± 1.8	83.8 ± 2.0	92.2 ± 1.3[1]
CIMDL [Wang et al., 2016]	87.3 ± 1.6	**84.2 ± 1.7**	92.4 ± 1.8
DCNN-GPC [Sun et al., 2017]	88.4 ± 2.1	80.3 ± 2.7	91.8 ± 1.1
STEM-CaRFs [Asif et al., 2017]	88.8 ± 2.0	80.8 ± 2.1	92.2 ± 1.3
Baseline Model [Aakerberg et al., 2017]	**89.5 ± 1.9**	**84.5 ± 2.9**	93.5 ± 1.1
This work - Ensemble A	-	-	93.7 ± 1.1
This work - Ensemble B	-	-	**94.2 ± 0.7**

dataset, and include these in the ensemble to possibly improve the performance even further.



(a) (b) (c) (d) (e)

Fig. 6: Examples of typical misclassifications. The first row shows images of the actual class. (a) 'Pitcher'→ 'Coffe mug', (b) 'Potato'→ 'Tomato', (c) 'Pear'→ 'Apple', (d) 'Mushroom'→ 'Garlic', (e) 'Peach'→ 'Sponge'.

VI. Conclusions

In this work, we have shown that forming an ensemble by combining the softmax probabilities of different complementary CNN based RGB-D object recognition models, with weighted averaging to create an aggregated prediction, increases the recognition performance compared to using a single baseline model. Our best performing ensemble has an accuracy of 94.2% on the Washington RGB-D Object Dataset, which to the best of the author's knowledge, is the highest accuracy ever reported on this dataset in the literature.

References

[Aakerberg et al., 2017] Aakerberg, A., Nasrollahi, K., and Heder, T. (2017). Depth value pre-processing for accurate transfer learning based rgb-d object recognition. *Under Review - IJCCI*.

[Asif et al., 2017] Asif, U., Bennamoun, M., and Sohel, F. A. (2017). Rgb-d object recognition and grasp detection using hierarchical cascaded forests. *IEEE Transactions on Robotics*, PP(99):1–18.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[Bonab and Can, 2016] Bonab, H. R. and Can, F. (2016). A theoretical framework on the ideal number of classifiers for online ensembles in data streams. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, pages 2053–2056, New York, NY, USA. ACM.

[Carlucci et al., 2016] Carlucci, F. M., Russo, P., and Caputo, B. (2016). A deep representation for depth images from synthetic data. *ArXiv e-prints*.

[Dalal and Triggs, 2005] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 886–893, Washington, DC, USA. IEEE Computer Society.

[Dietterich, 2000] Dietterich, T. G. (2000). Ensemble methods in machine learning. In *Proceedings of the First International Workshop on Multiple Classifier Systems*, MCS '00, pages 1–15, London, UK, UK. Springer-Verlag.

[Eitel et al., 2015] Eitel, A., Springenberg, J. T., Spinello, L., Riedmiller, M., and Burgard, W. (2015). Multimodal deep learning for robust rgb-d object recognition. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany.

[Goodfellow et al., 2013] Goodfellow, I. J., Mirza, M., Xiao, D., Courville, A., and Bengio, Y. (2013). An Empirical Investigation of Catastrophic Forgetting in Gradient-Based Neural Networks. *ArXiv e-prints*.

[Guo et al., 2014] Guo, Y., Bennamoun, M., Sohel, F., Lu, M., and Wan, J. (2014). 3d object recognition in cluttered scenes with local surface features: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(11):2270–2287.

[He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. *CoRR*, abs/1512.03385.

[Hinton et al., 2015] Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the Knowledge in a Neural Network. *ArXiv e-prints*.

[Jeff Donahue,] Jeff Donahue. BVLC CaffeNet Model.

[Jia et al., 2014] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc.

[Lai et al., 2011] Lai, K., Bo, L., Ren, X., and Fox, D. (2011). A large-scale hierarchical multi-view rgb-d object dataset. In *ICRA*, pages 1817–1824. IEEE.

[Lee et al., 2015] Lee, S., Purushwalkam, S., Cogswell, M., Crandall, D. J., and Batra, D. (2015). Why M heads are better than one: Training a diverse ensemble of deep networks. *CoRR*, abs/1511.06314.

[Li et al., 2015] Li, W., Cao, Z., Xiao, Y., and Fang, Z. (2015). Hybrid rgb-d object recognition using convolutional neural network and fisher vector. In *2015 Chinese Automation Congress (CAC)*, pages 506–511.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110.

[Razavian et al., 2014] Razavian, A. S., Azizpour, H., Sullivan, J., and Carlsson, S. (2014). CNN features off-the-shelf: an astounding baseline for recognition. *CoRR*, abs/1403.6382.

[Russakovsky et al., 2015] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556.

[Socher et al., 2012] Socher, R., Huval, B., Bath, B., Manning, C. D., and Ng, A. Y. (2012). Convolutional-recursive deep learning for 3d object classification. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 25*, pages 656–664. Curran Associates, Inc.

[Sun et al., 2017] Sun, L., Zhao, C., and Stolkin, R. (2017). Weakly-supervised DCNN for RGB-D Object Recognition in Real-World Applications Which Lack Large-scale Annotated Training Data. *ArXiv e-prints*.

[Szegedy et al., 2014] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. E., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2014). Going deeper with convolutions. *CoRR*, abs/1409.4842.

[Wang et al., 2016] Wang, Z., Lin, R., Lu, J., Feng, J., and Zhou, J. (2016). Correlated and individual multi-modal deep learning for RGB-D object recognition. *CoRR*, abs/1604.01655.