Multi-task Deep Learning for Image Understanding

Bo Yu
The State Key Laboratory of Remote Sensing Science, Institute of Remote Sensing and Digital Earth,Chinese Academy of Sciences,Beijing 100101, China
Graduate University of Chinese Academy of Sciences, Beijing 100049, China
Carnegie Mellon University
NASA Research Park #23, Moffett Field, CA 94043
Email: bo.yu@west.cmu.edu

Ian Lane
Carnegie Mellon University
NASA Research Park #23, Moffett Field, CA 94043
Email: lane@cs.cmu.edu

***Abstract*—Deep learning models can obtain state-of-the-art performance across many speech and image processing tasks, often significantly outperforming earlier methods. In this paper, we attempt to further improve the performance of these models by introducing multi-task training, in which a combined deep learning model is trained for two inter-related tasks. We show that by introducing a secondary task (such as shape identification in the object classification task) we are able to significantly improve the performance of the main task for which the model is trained. Using public datasets we evaluated our approach on two image understanding tasks, image segmentation and object classification. On the image segmentation task, we observed that the multi-task model almost doubled the accuracy of segmentation at the pixel-level (from 18.7% to 35.6%) compared to the single task model, and improved the performance of face-detection by 10.2% (from 70.1% to 80.3%). For the object classification task, we observed a 2.1% improvement in classification accuracy (from 91.6% to 93.7%) compared to a single-task model. The proposed multi-task models obtained significantly higher accuracies than previously published results on these datasets, obtaining 22.0% and 6.2% higher accuracies on the face-detetction and object classification tasks respectively. These results demonstrate the effectiveness of multi-task training of deep learning models for image understanding tasks.**

***Keywords*-image segmentation; deep learning;multi-task learning**

I. Introduction

Single-task learning is widely used in machine learning for image processing [1], [2]. 'Task' here represents the purpose we are learning. However, single-task learning focuses on the information of main purpose only, regardless of other related information. That would be more difficult to classify complex objects with various shapes, outlines, orientations and sizes [4] in the real world, such as face detection and object recognition.

3D information (color and depth) is a way to simplify complex object classification by adding distance to make the object of interest stereo. Also, depth data guides in detecting face or recognizing objects, especially in cases where images are rotated, overlapped, exposed to different illumination or even distorted by noise. The combination of depth information and 2D texture images is a promising method in improving recognition rates [5].

Mostly, methods in face recognition using 3D information are surface based. Vuong L. and Huang T.S. [8] represent each point in face with its corresponding facial level curve by calculating the distance between curves in the same level, which are classified by HMM (Hidden Markov Models). Colombo, A., Cusano C. and Schettini R. develop a method by curvature analysis in face detection [9]. Hg R.I., Jasek P., Rofidal C., Nasrollahi K. and Moeslund T.B. [10] compare the method of Colombo, et al [9] and PCA in face detection on their dataset. They obtained accuracies of 51.74% and 58.25% correspondingly. Similarly, methods used in 3D object recognition [11], [12] are mainly based on hand-crafted features. That requires strong analysis of object of interest. Moreover, the features extracted are arguable, as they are limited to different knowledge background.

Such limitations can be reduced by deep learning algorithms. Deep learning methods enhance performance in face recognition [2], facial key point detection [1] and object detection [13] by learning hierarchical features using raw data only. However, to the best of our knowledge, deep learning methods in face recognition based on both depth and 2D images have not received much systematic study. Furthermore, the proposed deep learning methods in image understanding are mainly single-task based, which may not work well for complex objects.

Multi-task learning is a multi-structure model, involving single-task and secondary-task. Single-task focuses on training using information of main application. Secondary-task, on the other hand, learns features from relative information, which can be anything related to our main purpose. For example, if we want to do face detection using multi-task model, relative information can be landmarks on the face. The combination of features learned from main and relative information can help improve accuracy in achieving main application [14]. Multi-task model has been applied to neural

978-1-4799-5934-1/14/$31.00 ©2014 IEEE

network for classification [14]. However, the network is shallow, and features extracted are not hierarchical.

This paper focuses on investigating the performance of mutli-task model in image understanding. The multi-task deep learning model in our paper is based on Convolutional Neural Network (CNN) [16] and Denoising Autoencoder (DA) [17]. It was applied to face detection and object recognition using 3D information (color and depth). A series of experiments, training single-task and multi-task respectively using 2D data, depth data and 3D data, were conducted. Their performances were evaluated by comparison with the state-of-art accuracies published on the datasets VAP [10] and RGB-D Object Dataset [15].

II. Network Architectures for Multi-task Deep Learning

A. Model Architecture

Our model is composed of two sub tasks. Single-task focuses on main purpose, while secondary-task works for something related. For face detection, the single-task is to classify each pixel into face or non-face, and secondary-task is to determine each pixel to be one of the landmarks on the face (eyes, nose, mouth, face skin) or non-face. In the case of object recognition, classifying each object into one of the categories is single-task. Classifying each object into one of four pre-defined shape categories can be selected as secondary-task to enhance the ability to distinguish different objects in single-task. Secondary-task supplements single-task by forcing multi-task to learn internal representation between main purpose and related one. To get most out of multi-task learning, we trained each task separately [14]. For both cases, secondary-task was first trained with its corresponding label to get supplementary features. Single-task was further trained on top of the parameters trained in secondary one. The classification labels of output layer in face detection are face and non-face, while in object recognition, labels are detemined by the categories in the dataset. In terms of the applications of our model, an example about our model for both purposes is in Figure 1.

Generally, the whole model is composed of secondary task (see Figure 1(c)) and single task (see Figure 1(b)). Secondary task consists of 6 layers (from $L0$ to $L5$). Single-task includes 7 layers (from $L0$ to $L6$). In addition, secondary task and single task share the same input layer $L0$. The combination of $L5$ in single task and secondary task forms the input of hidden layer $L6$. Finally, the output layer of the whole model is trained in single task. The first layer is the original image with width of W and height of H. $L1$ is the convoluted and pooled layer in both tasks. $L2$ is one dimensional, reshaped from $L1$. $L3$ is a hidden layer in both secondary and single tasks. $L4$ is the hidden layer of denoising auto-encoder to enhance the ability of our model in resisting noise and to decrease feature dimension. $L5$ is another hidden layer. Both sub-structures in layers $L4$ and $L5$ are the same.

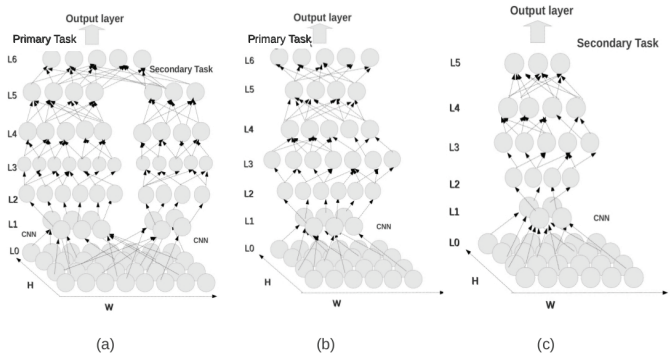

Figure 1. Architecture of the proposed multi-task.(a) is the structure of multi-task model (b) is single-task (c) represents secondary task

B. Training and optimization

We trained each sub task separately. The secondary-task was trained first. Optimized parameters trained for each layer were recorded. The single-task was then trained with the same training set. During single-task training, when it comes to $L5$ in each epoch, $L5$ in secondary-task was calculated using the optimized parameters trained previously. Values in $L5$ of single-task and secondary-task were then combined and used to generate $L6$. When doing back-propogation, parameters in secondary-task remain the same, only those in single-task were updated. To avoid overfitting, weight decay and early stopping were used. Early stopping outweighs the performance of regularization algorithms in many situations [19]. In our model, the stopping criteria is defined in Equations (4)-(6), according to the definition proposed by Prechelt [20]. The criteria is calculated using validation error, which is obtained from validation set. Validation set was randomly selected from training data, taking up 20%.

Criteria: stop when

$$\frac{GL(t)}{P_k(t)} > \alpha \tag{1}$$

where

$$GL(t) = 100 \times (\frac{E_{va}(t)}{E_{opt}(t)} - 1) \tag{2}$$

$$P_k(t) = 1000 \times (\frac{\sigma^t_{t'=t-k+1} E_{tr}(t')}{k \times min^t_{t'=t-k+1} E_{tr}(t')} - 1) \tag{3}$$

In Equations (4)-(6), $E_{opt}(t)$ is the lowest validation error obtained at epoch t. $E_{va}(t)$ represents validation error of epoch t. Moreover, $P_k(t)$ implies how much that average training error $\frac{\sigma^t_{t'=t-k+1} E_{tr}(t')}{k}$ is larger than the minimum value among the k epochs from epoch $t-k+1$ to t. By early stopping criteria, our training time is shortened. However, we still take the risk that early stopping may not work well

without a good definition of the criteria. Weight decay was used in our cost function [20], using a scale parameter of 0.003 [21].

To reduce the impact of possible unbalanced training data, we adopted probabilistic sampling in cost function [22]. The probabilistic sampling is a factor p_k that penalizes class k with its pattern number. p_k is defined in 4.

$$p_k = (1 - c_s)p_x + \frac{c_s}{N_c} \tag{4}$$

where N_c is the number of classes, c_s is a scaling factor, and p_x is the proportion of patterns in class k among all the patterns in training samples. Different learning rates were used. In single-task, the learning rate was 0.001, while for secondary-task, 0.01 worked well.

III. Experimental Evaluation

A. Datasets

Our model was evaluated in two application areas, face detection and object recognition. They are two of the most active areas in image processing. In face detection, the largest challenge is low detection rates in various poses and illumination conditions. For object recognition, different viewing angles and shapes for one category objects are the main obstable. To address such challenges, VAP (Visual Analysis of People Laboratory) RGB-D face dataset [10] and RGB-D Object Dataset [15] are used to evaluate our method.

VAP (Visual Analysis of People Laboratory) RGB-D face dataset [10] is one of the most recently published public dataset for 3D face identification and recognition. Its depth and RGB data are synchronized. 2D images are 1280x960 pixels and depth images 640x480 pixels. 31 persons are involved in collecting the dataset, including 17 face poses for each person.

RGB-D Object Dataset [15] provides a novel large scale of synchronized 2D and depth data, including 300 common turnable and rotated objects of 51 categories from different viewing angles. The objects are grouped using WordNet hyponym/hypernym relations. Totally, there are 250,000 3D images. Recently, many researchers have achieved impressive accuracies in object recognition [15], [23]–[25] on top of this dataset.

B. Pre-processing

For both datasets, all the 2D images were first transformed to YUV , because RGB is not perceptually uniform [26]. Next, both YUV and depth images were normalized by divisive constrast normalization (see Equation 5) [27] .

$$g_{norm}(i,j) = \frac{g_{origin}(i,j)^2}{S + \mu \times \sum_{m=i-k_h}^{m=i+k_h} \sum_{n=j-k_h}^{n=j+k_h} g_{origin}(i,j)} \tag{5}$$

where $g_{origin}(i,j)$ is the originl intensity of pixel (i,j). Correspondingly, $g_{norm}(i,j)$ is the normalized value of pixel (i,j). S and μ are scaling factors. k_h is the width of kernel to describe the size of local area. Divisive contrast normalization was adopted because it could reveal the local contrast of each pixel, rather than normalize all the pixel intensities of an image to a specific scale only. It is more suitable for our case since local information plays a key role in describing different subjects.

C. Experiment for face detection

Depth data in VAP [10] were synchronized to 2D images by the method introduced in [28]. Finally, all the depth and YUV matrices were downsampled into 320x240 pixels by linear interpolation [29]. Owing to the fact that our model for face detection is pixel-based image segmentation, a sub-region in the size of 51x51 centered at each pixel in the image was generated and used as a sample to do face detection. Each sub-region was asigned by the label of its center pixel. To compare with published results [9], [10], in which testing was conducted for each pose, we selected 3 images from different people for each pose to generate training data. For each selected image, 51x51 sub-regions were generated for training, centered by all the pixels. Therefore, we had 3,234,675 trainig samples. Of all the training data, 20% of which were randomly selected as validation set to calculate early stopping criteria value in Equations (4)-(6). Test data was composed of all the other images in VAP. Detected faces (with blue bounding rectangles) of samples in Figure **??** are correspondingly shown in right column.

1) Experimental Setup: To analyze performance of our model on VAP in detail, we conducted six experiments in total: (1) Single-task model using 2D data (S-C); (2) Single-task model using depth data (S-D); (3) Single-task model using 3D data(S-CD); (4) Multi-task model using 2D data (M-C); (5) Multi-task model using depth data (M-D); (6) Multi-task model using 3D data (M-CD) (see Table I). In our model structure(see Figure 1), $L0$ is 51x51x4 pixels (4 represents 4 channels, Y,U,V and Depth). The filter size in single-task is 36x36 pixels and 46x46 in secondary. $L2$ is in size of 1080 and 7680 respectively in single and secondary tasks. $L3$ is 1000 in both tasks. $L4$ decreases the feature size from 1000 to 500. $L5$ also reduces the feature size from 500 to 300. A bounding rectangle was used to mark the face (Figure 2).

2) Results and analysis: Visually, we can see that faces detected by models other than multi-task using 3D data are almost the same (see Figure 2). Their bounding boxes take similar shape and position. Nonetheless, faces detected by mutli-task using 3D data are more practical, with less pixels misclassified as faces. To evaluate the performance of our algorithm more objectively and statistically, detection rates of each pose among all the data were calculated in the six

Table I
EXPERIMENTAL CONDITIONS

ID	Abbreviation	Model-type	Features
1	S-C	Single-task	color
2	S-D	Single-task	depth
3	S-CD	Single-task	color+depth
4	M-C	Multi-task	color
5	M-D	Multi-task	depth
6	M-CD	Multi-task	color+depth

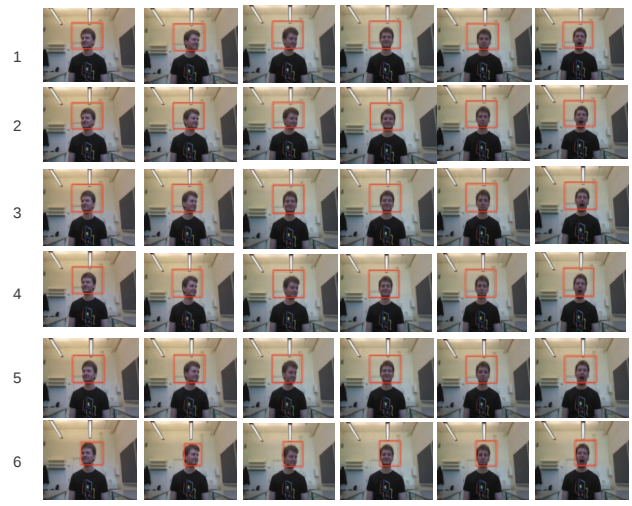

Figure 2. Faces detected by six models. As shown in Table I, '1' represents faces detected by S-C. '2' is faces detected by S-D. '3' is by S-CD. '4', '5' and '6' are correspondingly faces detected by M-C, M-D and M-CD.

experiments from (1)(S-C) to (6)(M-CD). We divided the performance evaluation into two parts. One is multi-task model vs. single-task and two other published results (see Table II). The other is using 3D vs. 2D or depth data (see Table III).

Multi-task model vs. other model

From Table II we can see our model substantially improved the accuracy of face detection on the dataset of VAP by more than 20%, compared with FACE Triangles detection (F-T) [9] and PCA (from [10]). Moreover, the accuracy of multi-task outweighs that of single task by almost 10% higher. This indicates that secondary-task, together with a shared representation, helps learn features more accurately. Nevertheless, all the methods show a poor performance when people look downward (see Figure 3). That possibly owes to much shadow overlaps face when posing downward, which hinders detecting face. Morever, PCA fluctuates with different pose markedly, and F-T performs smoothly but obtains the lowest or the second lowest detection rates. Statistically, multi-task model significantly improved the detection accuracy (at 95% confidence level), compared with baselines F-T (99.8%>95%), PCA (99.7%>95%) using Chi-square.

Table II
ACCURACY(%) OF DETECTION RATES ON VAP DATASET BY MULTI-TASK MODEL, SINGLE-TASK MODEL, FACE TRIANGLES DETECTION(F-T) [9] AND PCA(FROM [10])

Data	F-T	PCA	S-AVE	M-AVE
Overall accuracy	51.7	58.3	70.2	80.3

Figure 3. Detection rates of four methods on VAP. VAP are grouped into 8 groups according to the pose directions and different expressions. 'Downward' being seperated from other poses owes to the reason that our method performs poor under this situation. The detection rates of 'Upward', 'Horizontal' and 'Slight down' are the average values of each pose indexed 1-4,5-9,10-12 in VAP seperately.

Table III
ACCURACY OF DETECTION RATES ON VAP DATASET BY SINGLE-TASK(S) AND MULTI-TASK(M) USING COLOR (C), DEPTH (D), COLOR-DEPTH (CD) DATA SEPERATELY(%)

Method	C	D	CD
Single-task	66.2	66.3	70.2
Multi-task	75.4	75.2	80.3

3D vs. 2D or depth data 3D data work better than color or depth data only. Still, models using 3D data perform substantially better than that using 2D data or depth data (see Table III) . In addition to detection rates, segmentation accuracy of faces detected in bounding boxes are calculated according to Equation 6 [30]. d_i and l_i are detected and annotated face regions respectively. Multi-task model achieves better segmentation accuracy than single-task model (see Table IV), which agrees with the observations in our previous section study. Further experiments show that M-CD not only significantly (at 95% confidence level, using Chi-square) outperforms M-C (99.998%>95%), M-D (99.998%>95%), S-CD (99.998%>95%), S-C (99.995%>95%) or S-D (99.998%>95%) in terms of segmentation accuracy, but also improves significantly in detection rates (M-C: 99.999%>95%, M-D: 99.999%>95%, S-CD: 99.998%>95%, S-C: 99.999%>95% and S-D: 99.999%>95%). Consequently, multi-task model using 3D data can be used to detect face more practically and accurately.

$$S(d_i, l_i) = \frac{area(d_i) \cap area(l_i)}{area(d_i) \cup area(l_i)} \quad (6)$$

Table IV
ACCURACY OF SEGMENTATION USING S-C,S-D,S-CD,M-C,M-D,M-CD AT PIXEL LEVEL(%)

Method	C	D	CD
Single-task	17.9	17.8	18.7
Multi-task	19.1	19.5	35.6

Figure 4. Four general shapes of objects.

Table V
ACCURACY(%) OF OBJECT RECOGNITION ON RGB-D OBJECT DATASET.CD IS SHORT FOR COLOR-DEPTH DATA.

Method	C	D	CD
Lai *et al.* [15]	74.5	64.7	83.8
Lai *et al.* [25]	78.6	70.2	85.4
Bo *et al.* [24]	80.7	80.3	86.5
Bo *et al.* [23]	82.4	81.2	87.5
Single-task model	90.8	85.3	91.6
Multi-task model	92.3	92.4	93.7

Table VI
PERFORMANCE(%) OF OBJECT RECOGNITION ON RGB-D OBJECT DATASET.

Method	Confidence interval
Linear SVMs [15]	[79.1-84.7]
Nonlinear SVMs [15]	[80.3-87.3]
Random Forest [15]	[75.6-83.6]
Combination of all HKDES [24]	[81.9-86.3]
Multi-task using color-depth	[89.9-94.3]

D. Experiment for object recognition

Unlike segmentation, object recognition needs a whole image as input. Therefore, all the data in RGB-D Object Dataset were resized to 51x51 pixels. The secondary-task uses shape character of objects in building multi-task model. Among the 250,000 color-depth images in the dataset, 41,877 color-depth images were used as testing data. That was suggested by $http : //rgbd - dataset.cs.washington.edu/dataset/rgbd - dataset_eval/$. 20% of the rest dataset were used for validation, and the others for training. Similar to the experiment before, we used 6 combinations of single-task, mutli-task using depth, color and color-depth data to do object recognition. The corresponding recognition rates and state-of-art results are shown in Table V.

It is worth noting that multi-task model using 3D data performs the best compared with state-of-art methods on this dataset. In terms of using 2D or depth data, multi-task achieves 10% higher accuracy than the algorithm proposed in [23]. In addition, the performance of multi-task using 3D data outweighs that of [23] (Table V). On top of that, statistical analysis indicates that multi-task model using 3D data improves the performance significantly(at 95% confidence level, using t-test) compared with recently proposed baseline performances as well(see Table VI).

IV. CONCLUSION

Designing hand-crafted features is difficult and time demanding. Single task model learns monotonous features, which conveys relative information and can not fully represent features of different objects. Our work focuses on resolving such difficulties by applying multi-task model in deep learning methods for learning more synthetic features. Our results indicate that deep learning based multi-task model can be used to improve recognition and detection rates in various image processing applications markedly.

ACKNOWLEDGMENT

The authors are grateful to Bing Liu's generous help from Carnegie Mellon University in improving language quality of our paper.

REFERENCES

[1] S. Yi, W. Xiaogang, and T. Xiaoou, "Deep Convolutional Network Cascade for Facial Point Detection," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013, pp. 3476–3483.

[2] W. Yue, W. Zuoguan, and J. Qiang, "Facial Feature Tracking Under Varying Facial Expressions and Face Poses Based on Restricted Boltzmann Machines," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013, pp. 3452–3459.

[3] M. Osadchy, Y. L. Cun, and M. L. Miller, "Synergistic Face Detection and Pose Estimation with Energy-Based Models," *The Journal of Machine Learning Research*, vol. 8, pp. 1197–1215, 2007.

[4] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[5] K. Chang, K. Bowyer, and P. Flynn, "An evaluation of multimodal 2D+3D face biometrics," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 4, pp. 619–624, 2005.

[6] F. Tsalakanidou and S. Malassiotis, "Real-time 2D+3D facial action and expression recognition," *Pattern Recognition*, vol. 43, no. 5, pp. 1763–1775, 2010.

[7] F. Tianhong, Z. Xi, S. Shah, and I. Kakadiaris, "4D facial expression recognition," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, 2011, pp. 1594–1601.

[8] L. Vuong, T. Hao, and T. Huang, "Expression recognition from 3D dynamic faces using robust spatio-temporal shape features," in *Automatic Face Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, 2011, pp. 414–421.

[9] A. Colombo, C. Cusano, and R. Schettini, "3d face detection using curvature analysis," *Pattern recognition*, vol. 39, no. 3, pp. 444–455, 2006.

[10] R. Hg, P. Jasek, C. Rofidal, K. Nasrollahi, T. Moeslund, and G. Tranchet, "An RGB-D Database Using Microsoft's Kinect for Windows for Face Detection," in *Signal Image Technology and Internet Based Systems (SITIS), 2012 Eighth International Conference on*, 2012, pp. 42–46.

[11] S. Berretti, N. Werghi, B. A. Del, and P. Pala, "Geometric histograms of 3D keypoints for face identification with missing parts," in *Eurographics 2013 Workshop on 3D Object Retrieval*. The Eurographics Association, 2013, pp. 57–64.

[12] Y. Lei, M. Bennamoun, M. Hayat, and Y. Guo, "An efficient 3d face recognition approach using local geometrical signatures," *Pattern Recognition*, vol. 47, no. 2, pp. 509–524, 2014.

[13] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 233–240.

[14] R. Caruana, "A Dozen Tricks with Multitask Learning," in *Neural Networks: Tricks of the Trade*, ser. Lecture Notes in Computer Science, G. Orr and K.-R. Müller, Eds. Springer Berlin Heidelberg, 1998, vol. 1524, pp. 165–191.

[15] 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*. IEEE, 2011, pp. 1817–1824.

[16] Y. LeCun, L. Jackel, L. Bottou, A. Brunot, C. Cortes, J. Denker, H. Drucker, I. Guyon, U. Muller, E. Sackinger *et al.*, "Comparison of learning algorithms for handwritten digit recognition," in *International conference on artificial neural networks*, vol. 60, 1995.

[17] G. E. Hinton and R. R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[18] C. Minmin, E. X. Zhixiang, Q. W. Kilian, and S. Fei, "Marginalized Denoising Autoencoders for Domain Adaptation," in *Proceedings of the 29th International Conference on Machine Learning, Edinburgh, Scotland, UK, 2012.*, vol. abs/1206.4683, 2012.

[19] W. Finnoff, F. Hergert, and H. G. Zimmermann, "Improving model selection by nonconvergent methods," *Neural Networks*, vol. 6, no. 6, pp. 771–783, 1993.

[20] L. Prechelt, "Early Stopping - But When?" in *Neural Networks: Tricks of the Trade*, ser. Lecture Notes in Computer Science, G. Orr and K.-R. Müller, Eds. Springer Berlin Heidelberg, 1998, vol. 1524, pp. 55–69.

[21] D. C. Plaut, S. J. Nowlan, and G. E. Hinton, "Experiments on learning by back propagation," 1986.

[22] Y. Larry, L. Richard, and W. Brandyn, "Effective Training of a Neural Network Character Classifier for Word Recognition," 1997.

[23] L. Bo, X. Ren, and D. Fox, "Unsupervised feature learning for rgb-d based object recognition," *ISER, June*, 2012.

[24] L. Bo, K. Lai, X. Ren, and D. Fox, "Object recognition with hierarchical kernel descriptors," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1729–1736.

[25] K. Lai, L. Bo, X. Ren, and D. Fox, "Sparse distance learning for object recognition combining rgb and depth information," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 4007–4013.

[26] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *Acm Computing Surveys (CSUR)*, vol. 38, no. 4, p. 13, 2006.

[27] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?" in *Computer Vision, 2009 IEEE 12th International Conference on*, 2009, pp. 2146–2153.

[28] Y. Lijun, W. Xiaozhou, S. Yi, W. Jun, and M. Rosato, "A 3D facial expression database for facial behavior research," in *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*, 2006, pp. 211–216.

[29] E. Meijering, "A chronology of interpolation: from ancient astronomy to modern signal and image processing," *Proceedings of the IEEE*, vol. 90, no. 3, pp. 319–342, 2002.

[30] V. Jain and E. G. Learned-Miller, "Fddb: A benchmark for face detection in unconstrained settings," *UMass Amherst Technical Report*, 2010.