Enlightening Deep Neural Networks with Knowledge of Confounding Factors

Yu Zhong Gil Ettinger
{yu.zhong, gil.ettinger}@stresearch.com
Systems & Technology Research

Abstract

Deep learning techniques have demonstrated significant capacity in modeling some of the most challenging real world problems of high complexity. Despite the popularity of deep models, we still strive to better understand the underlying mechanism that drives their success. Motivated by observations that neurons in trained deep nets predict variation explaining factors indirectly related to the training tasks, we recognize that a deep network learns representations more general than the task at hand in order to disentangle impacts of multiple confounding factors governing the data, isolate the effects of the concerning factors, and optimize the given objective. Consequently, we propose to augment training of deep models with auxiliary information on explanatory factors of the data, in an effort to boost this disentanglement. Such deep networks, trained to comprehend data interactions and distributions more accurately, possess improved generalizability and compute better feature representations. Since pose is one of the most dominant confounding factors for object recognition, we adopt this principle to train a pose-aware deep convolutional neural network to learn both the class and pose of an object, so that it can make more informed classification decisions taking into account image variations induced by the object pose. We demonstrate that auxiliary pose information improves the classification accuracy in our experiments on Synthetic Aperture Radar (SAR) Automatic Target Recognition (ATR) tasks. This general principle is readily applicable to improve the recognition and classification performance in various deep-learning applications.

1. Introduction

In recent years, deep learning technologies, in particular Deep Convolutional Neural Networks (DCNNs) [33], have taken the computer vision field by storm, setting drastically improved performance records for many real world computer vision challenges including general object recognition [9][18][30][45], face recognition [51], scene classification [62], object detection and segmentation [15] [22], feature encoding [28], metric learning [24], and 3D reconstruction [11]. The dominantly superior performance by deep learning relative to other machine learning approaches has also emerged in numerous other application fields – including speech recognition and natural language processing – generating unprecedented enthusiasm and optimism in artificial intelligence in both the research community and the general public. This overwhelming success of deep learning is propelled by three indispensable enabling factors:

1. Groundbreaking algorithm developments in exploitation of deep architectures and effective optimization of these networks, allowing capable representation and modeling of complex problems [20][30][33];
2. Availability of very large-scale training datasets that capture full data variations in real world applications in order to train high capacity neural networks [8]; and
3. Advanced processing capability in graphical processing units (GPU) enabling computation in speed and scale that was impossible earlier.

Despite the sweeping success of DCNNs, we still strive to understand how and why they work so well in order to better utilize and master them. In this paper we contemplate what deep networks have actually learned once they have been trained to perform a particular task and explore how to take advantage of that knowledge. Based on observations reported in multiple research efforts that neurons in trained deep networks predict attributes that are not directly associated with the training tasks, we perceive that deep models learn structures more general than what the task at hand involves. This generalizability likely results from performance optimization on data populations with multiple explanatory factors, which is typical in real world applications. As a result, we can assist the unsupervised learning of latent factors that naturally occur in the training of deep neural networks, by supplying supervisory signals on dominant confounding factors during training. Information on such data impacting factors allows the network to untangle data interactions, isolate the impact of the factors of interest, learn more accurate characterization of the underlying data distributions, and generalize better with new data.

With this principle, we propose to augment the training of DNNs using information on confounding factors in order to improve their performance. We describe a general framework to boost training of any standard deep architecture with auxiliary explanatory factors that account for significant data variations. Such information has often been overlooked because it is deemed irrelevant to the task at hand. Nonetheless, it can help reducing ambiguity in the data and aid in classification and recognition. We apply the proposed framework to build a pose-aware DCNN for object recognition by injecting pose information in addition to class labels during training to improve the classification accuracy of the neural network.

In this paper we make the following contributions.

- We describe a general framework to augment training of DCNNs using available information on influential confounding factors of the data population. This framework can be applied to any existing deep architecture at a very small additional computational cost.
- To verify this finding we apply the principle to augment existing DCNNs and demonstrate performance gains using real world data sets. To address pose variations in object recognition, we train a novel pose-aware DCNN architecture by explicitly encoding both pose and object class information during training and demonstrate the auxiliary pose information indeed increases the classification accuracy.

The remainder of the paper is organized as follows. We review related literature in Section 2 and motivate our approach in Section 3. Section 4 describes a general framework to take advantage of auxiliary explanatory factors to improve the performance of DCNNs. We describe how to train a pose-aware DCNN for recognition tasks and present related experiments in Section 5. We draw conclusions in Section 6.

2. Literature Review

DCNNs have demonstrated unmatched capability to tackle very complex challenges in real world applications. Understanding the fundamentals of DNNs helps us to better utilize them. We review techniques that improve the performances of deep networks.

The capacity of a DCNN can be increased by either expanding its breadth to have more feature maps at each layer [54], or by growing the depth of the network [50]. As deep models demand an enormous amount of training data to offset the risk of over-fitting, data augmentation improves the accuracy of the trained deep models [30]. As deeper architectures become more difficult to optimize, auxiliary classifiers at intermediate layers help to flush gradient flow to lower layers during back-propagation and improve training performance [50]. DenseNets [25], residual networks [19], and highway networks [46] have been proposed to effectively optimize extremely deep networks. Nonlinearity such as Rectified Linear hidden Units (ReLU) is a major factor that enables deep networks to encode complex representations [7]. Variants of ReLUs have also been proposed [18] [34].

Hinton et al. used “drop-out” to prevent over-fitting due to co-adaptation of feature detectors by randomly dropping a portion of feature detectors during training [20]. Dropout training can be considered as a form of adaptive regularization to combat over-fitting [53]. A “maxout” network was subsequently proposed to improve both the optimization and accuracy of networks with dropout layers [16]. An alternative regularizer is batch normalization [27] that integrates normalization of batch data as a part of the model architecture and performs normalization for each training mini-batch to counter the internal covariate shift during training.

Multitask learning [3] trains several related tasks in parallel with a shared representation where what is learned for each task helps in learning other tasks. It is argued that extra tasks serve as an inductive bias to improve the generalization of the network. [6] used a single deep network to perform a full list of similar NLP tasks including speech tagging, parsing, name-entity recognition, language model learning, and semantic role labeling. [61] proposed to optimize facial landmark detection with related tasks such as categorical head pose estimation. Recently, multitask DCNNs have been successfully used to simultaneously perform multiple tasks including depth/surface normal prediction and semantic labeling [11], object detection and segmentation [15][17], and object detection, localization, and recognition [44]. In [14] and [31], multitask learning was used to address the correlations in object classes and emulating the hierarchical structure in object categorizations. Despite the successes of multitask learning, challenges remain to better understand how the mechanism works and to determine what kind of tasks help each other [3].

Real world applications often involve complex data arising from various sources and their interactions. It is fundamental to disentangle the factors of variation for many AI tasks [2]. There have been emerging efforts to discover and separate these factors in unsupervised or semi-supervised learning, such as generative models, where accurate data modeling and reconstruction demand knowledge of data explanatory factors. [35] used adversarial training with autoencoders to learn complementary hidden factors of variations. A cross-covariance loss was introduced in a semi-supervised autoencoder to learn factors of data variation beyond the observed labels [5]. InfoGan was proposed to disentangle factors fully unsupervised by maximizing mutual information between latent variables and observations [4]. Deep Convolution Inverse Graphics Network further coupled interpretable data transforms and latent variables by clamping images of specific transformations to the learning of latent variable intended for such transforms [32]. Since object pose is a major source of variation, [56] used a recurrent convolutional encoder-decoder network to disentangle pose and identity and synthesize new views.

Despite of increasing interests in disentangling factors of variation for unsupervised learning, less attention has been focused on their importance for supervised learning. A discriminative network often relies only on labels for the classification task and discards other informational sources of variation beneficial to data understanding. Our work fills the gap to explore the use of such factors of variation to improve DCNN classification performance.

While multitask deep networks have become more popular than ever, the choice of related tasks are usually ad hoc. It remains a major open problem to "better characterize, either formally or heuristically, what *related* tasks are" for multitask learning [3]. The proposed work, which uses auxiliary tasks related to prominent factors of data variation, sheds insight on the favorable choice of tasks for multitask deep learning and helps to better understand the underling mechanism for multitask deep networks. For example, it is possible tasks become related and beneficial to each other when the data observations from different tasks share common explanatory factors. Consequently, it is possible to find a set of explanatory factors that explains away enough amounts of data variations to achieve performance gains obtained from more complex auxiliary tasks.

3. Motivation

Deep convolutional neural networks distinguish themselves from traditional machine learning approaches in enabling a hierarchy of concrete to abstract feature representations. A study on performance-optimized deep hierarchical models trained for object categorization and human visual object recognition abilities indicates the trained network's intermediate and top layers are highly predictive of neural responses in the visual cortex of a human brain [55]. It has been suggested that the strength of DCNNs comes from the reuse and sharing of features, which results in more compact and efficient feature representations that benefit model generalization [1][2]. For example, the same convolutional filter bank is learned for the entire image domain in a DCNN, as opposed to learning location dependent filters.

An intriguing aspect of DCNNs is their remarkable transferability [12] [57]. A deep network trained on one dataset is readily applicable on a different dataset. The ImageNet model by Zeiler and Fergus [58] generalizes very well on the Caltech datasets. In other works, deep models trained to perform one task, such as object recognition, can be repurposed to significantly different tasks, such as scene classification, with little effort [9][21]. These natural generic modeling capabilities across tasks have also been demonstrated in the success of several integrated deep convolutional neural networks proposed to simultaneously perform multiple tasks for various applications [11][15][17][37][44]. Such facts indicate deep networks learn feature representations more pertinent about the data population than a specific task requires.

Furthermore, there has been significant empirical evidence emerging in the latest research that the neurons in trained DCNNs actually encode information that is not directly related to the training objectives or tasks. Semantic segregation of neuron activations on attributes such as "indoor" and "outdoor" has occurred in deep convolutional networks trained for object recognition, which has prompted application of these "DeCAF" [9] features to novel generic tasks such as scene recognition and domain adaptation with success. On the other hand, Zhou et al. [62] noticed that their DCNN trained for scene classification automatically discovered object categories relevant to the scene categories, even though only scene labels were used in training. Khorrami et al. [29] observed that deep neural networks that are trained for face recognition actually learn facial actions in some of its hidden neurons. The DeepID [48][49] network trained for face identification predicts gender information even though only identity labels are used in training. These observations suggest that deep models learn not only compact and reusable feature representations for the tasks that they are trained on, but also information more general and fundamental in order to optimize performance.

Data populations in real applications encompass wide ranges of variations. Some are due to the factors of concern, while others are not. It is usually impossible to address one factor in isolation without taking into consideration some of the other confounding factors. It is therefore necessary to either develop invariant features or explicitly model the effects of the confounding factors that significantly impact the data. For example, for face recognition applications, a face image depends on not only the identity, but also various other nuisance factors such as pose, lighting, facial expression, age, etc. An accurate face identification system needs to factor out the effects of these confounding factors.

When a DCNN is trained using a large data population to perform one specific task, it may naturally organize the network to capture the intrinsic data distribution governed by multiple influential explanatory factors. That is, as a result of the objective optimization, it needs to explain away the effects of the confounding factors. This explains the phenomenon in which neurons that predict auxiliary attributes arise in DCNNs trained for unrelated tasks.

If unsupervised learning of latent factors naturally occurs during the training of a DCNN for a specific task, boosting the training with information on influential auxiliary variables is then expected to help reduce error and better capture the underlying data distribution for increased generalization power. This motivates us to investigate deep architectures that take advantage of available information on explanatory factors for improved prediction performance.

In the following sections we describe augmented training of DCNNs using dominant confounding factors. We then instantiate a pose-aware deep network for general object recognition using this principle, and evaluate its performance on a SAR automatic target recognition task.

4. Augmented Training Using Confounding Factors

Knowledge of auxiliary explanatory factors can be easily incorporated into the deep learning framework as separate output constraints or losses in addition to the original outputs, in a manner similar to multiple outputs in multi-task deep networks [3][6]. A deep network trained this way is more comprehensive and knowledgeable.

These additional constraints during training limit the solution space of the network. The constraints introduced by a dominant confounding factor likely shape the solution space in a principled way to reflect the impact of this factor on the underlying data distribution. In another perspective, this training augmentation of factors of variation can be considered as a form of regularization, in the sense that "the basic idea of all regularization methods is to restrict the space of possible solutions" [40]. This regularization influences the network to more accurately capture the structure due to multiple explanatory factors and their interactions, and consequently improves the network's generalizability and performance.

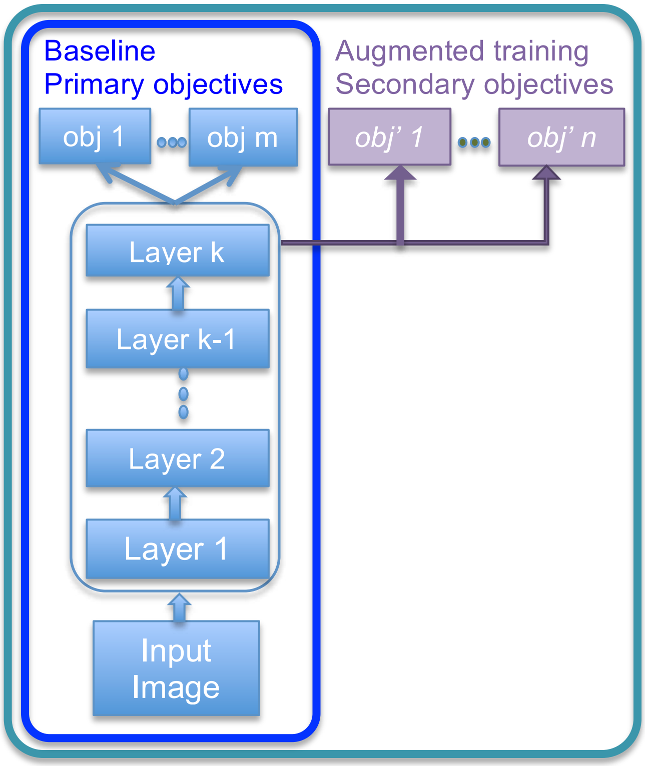


Figure 1. General framework to augment training of deep convolutional neural networks. On the left is a standard (baseline) deep convolutional neural network (shown in blue) with one or more objectives. We augment the training using influential auxiliary data explanatory variables as secondary prediction blocks (shown in green). Knowledge of these confounding variables shapes the weights of the network via gradient back-propagation originating from these secondary prediction blocks.

Figure 1 illustrates the general framework to augment training of deep convolutional neural networks. We start with a conventional architecture, consisting of convolutional layers at the bottom, then full-connected layers, and one or more prediction blocks at the top. Although the shown baseline architecture takes the simplest linear form, it can potentially be any of the existing deep architectures.

We then introduce additional objective blocks that take input from the existing top or intermediate hidden layers to predict one or more auxiliary confounding variables. During training, the network optimizes a weighted sum of primary objectives for the original task, and secondary objectives reflecting additional knowledge pertinent to the data distribution. The secondary objectives can be custom designed for each factor, and their weights reflect the importance and variation of these factors, which can be application dependent. The circuits to optimize the secondary objectives are fairly small compared to the remaining network. Information of confounding factors for the training data is infused to shape the network via these secondary prediction blocks during training, so that a DCNN learns and encodes both primary variables and secondary confounding factors of the input data, with a negligible cost in memory and computation time.

Once the network has been trained, the auxiliary circuits for secondary factors can be removed to obtain a model with exactly the same architecture as the baseline, in terms of number of layers and number of neurons at each layer. However, the parameter values learned with the auxiliary information make this DCNN more comprehensive and discerning to perform accurate classification during testing at no additional computational cost.

Note that this general framework fully observes the principle of compact and reusable feature representation, a major strength of deep convolutional neural networks [1][2]: a single network is employed to model various factors and interactions in the input data; feature representations are shared to fulfill various objectives. We simply supply the network with relevant information regarding the data and let the DCNN optimize for the best performance.

In the following section we apply the general framework to introduce a pose-aware DCNN to perform enhanced object recognition robust to pose variations.

5. Pose-Aware Deep Convolutional Neural Networks for Object Recognition

Pose is one of the most dominant confounding factors for recognition tasks. The same object can have drastically different appearances when the viewpoint varies. It is essential for a high performing object recognition system to address such variations [47][59].

Since variations introduced by pose are systematic, knowing the pose helps to explain away the induced variation for more accurate recognition. We investigate boosting DCNN classification performance using auxiliary pose information. For this purpose, our baseline architecture has a single prediction block at the top for classification. We introduce a secondary prediction block that takes input from the top hidden layer to regress on the pose variables. The secondary circuit introduced contains only $(N(k)+1)\times p_d$ parameters, where $N(k)$ is the number of outputs from the top hidden layer, and p_d is the dimension of the pose variable.

We train the pose-aware DCNN using both class labels and pose information to optimize a weighted sum of two objectives: one on the predicted class error of the inputs (Eq **1**), and the other on the pose alignment error (Eq **3**).

We use the popular softmax log-loss function for the classification task to minimize the class prediction error:

Eq 1: $\boldsymbol{obj^{CLASS}(net, X) = -\sum_{i\in\mathfrak{I}}(x_{ic_i} - log\sum_{j=1}^{C} e^{x_{ij}})}$,
where ***net*** stands for parameters of the network, ***X*** represents the training data, $\mathfrak{I}$ is the training set, *C* is the number of classes, x_{ij} is the response of the *i*-th input for the *j*-th class, and x_{ic_i} is the response of the *i*-th input for its truth class c_i.

Unlike class labels, pose variables are usually continuous. We perform regression using outputs from the top hidden layer to predict object pose. We only consider rotation here, as translation can be bypassed by either centering the image at object center, or by augmenting training data using randomly translated training images to achieve invariance to translation. In the case that the translation needs to be explicitly modeled, it is straightforward to add it into the formulation as well.

We represent 3D rotations using quaternions [23] for their desirable properties when compared to Euler angles or rotation matrices. We compute the distance between two rotations q_1 and q_2 as follows, which is both boundedly equivalent to the geodesic distance between the two rotation quaternions on the unit sphere, and converging fast as it approaches zero [26]:

Eq 2: $\boldsymbol{dist(q_1, q_2) = \arccos(|q_1 \cdot q_2|)}$

where the range is $[0, \frac{\pi}{2}]$. This distance function is pseudo-metric on the unit quaternion but is a metric function on the special orthogonal group $SO(3)$ of orthogonal matrices with determinant 1. The loss function for pose regression is the sum of the distance between the rotation predicted for each training image and its truth rotation:

Eq 3: $\boldsymbol{obj^{POSE}(net, X) = \sum_{i\in\mathfrak{I}} dist(\hat{q}_i, \tilde{q}_i)}$,
where $\hat{q}_i$ and $\tilde{q}_i$ are the quaternions for the predicted pose and the truth pose of the *i*-th training input respectively.

During training, the following combined objective function is minimized to learn the parameters of a deep network that simultaneously minimizes errors in predicted class labels and object poses of an input:

Eq 4: $\boldsymbol{obj^{COMBO}(net, X) =}$
$\boldsymbol{obj^{CLASS}(net, X) + \lambda \cdot obj^{POSE}(net, X)}$

Even though the ultimate goal of the network is to perform classification, the auxiliary pose information helps the network to disentangle confounding factors that influence the input data and better characterize the categorical traits.

We have applied our pose-aware DCNN architecture to improve automatic target recognition accuracy on SAR chips [13][41][42], an area where deep learning has recently demonstrated significant performance gains. Morgan first used a basic deep convolutional neural network [36] for SAR ATR. Later, “A-ConvNets” [54] was proposed to address the issue of over-fitting by replacing the fully connected layers in conventional deep neural networks with convolutional layers of local support

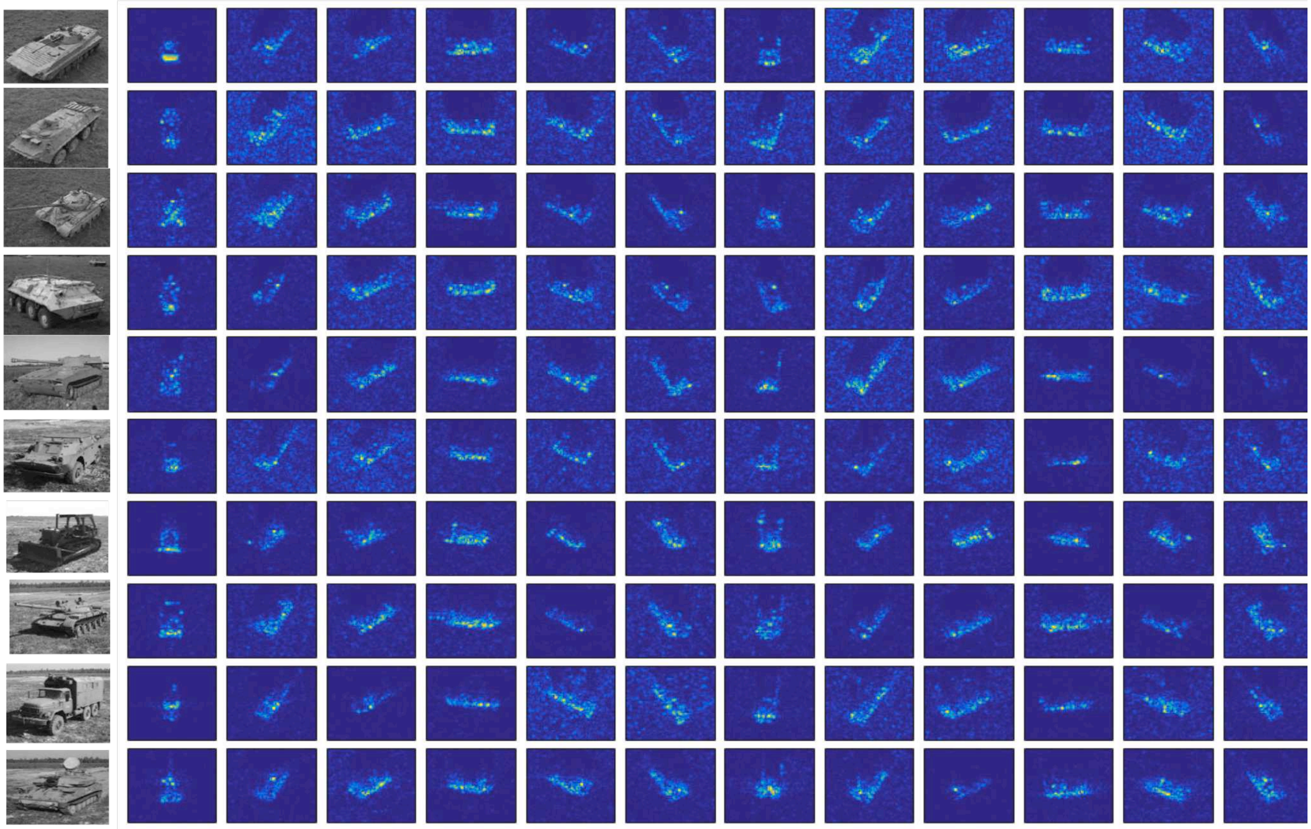


Figure 2. Targets in the public MSTAR SAR dataset of ten target classes, from top to bottom, bmp2, btr70, t72, btr60, 2S1, brdm2, d7, t62, zil131, and zsu23-4. Each row shows an example picture of the target class, followed by SAR chips for this target class, in the order of increasing azimuth angle, from 0° to 360°. It is evident that azimuth angle drastically affects the appearance of the SAR chips. For each SAR chip, the vertical axis corresponds to range, and the horizontal axis corresponds to cross-range.

to scale back the number of parameters for the deep network. Both approaches used only the target class labels in training the networks.

We have used the publicly available MSTAR dataset [42], the standard benchmark for SAR ATR. This dataset contains SAR chips for ten target classes, with sample images and SAR chips of the targets shown in Figure 2. Note that the azimuth angle of the target drastically affects the appearance, since different radar scattering structures are illuminated as the target rotates relative to the sensor. We have followed the common convention to partition the training/testing sets using depression angles [42], producing a total of 6,073 training images and 5,378 test images. Noticing the symmetry w.r.t the range axis in most of the targets, as shown in Figure 2, we have applied horizontal flip augmentation to double the size of the training dataset. To train the pose-aware DCNN, we also associate each flipped image with a pose. Since the azimuth angle is 0 when a target is head on, we negate the azimuth angle of the original image and assign it to its flipped image. For this dataset of moderate size, we have used three convolutional layers followed by one fully connected layer, as shown in Figure 3.

Since all targets are on the ground plane, the 3D pose of a target degenerates to the azimuth angle of the target w.r.t the sensor. For this special case, Eq **2** of the distance between two poses becomes

Eq 5: $dist(q_1, q_2) = \arccos(\left|\cos((\theta_1^{azimuth} - \theta_2^{azimuth})/2)\right|)$

which has a range of $[0, \frac{\pi}{2}]$.

We follow common practices [30][52] to train the network once the objective function is defined. The weights and parameters of the network are randomly initialized with zero mean Gaussian distributions with $\sigma = 0.01$. Stochastic Gradient Descent is used in conjunction with momentum and weight decay with a mini-batch size of 100. The other parameter values used in training are shown in Table 1.

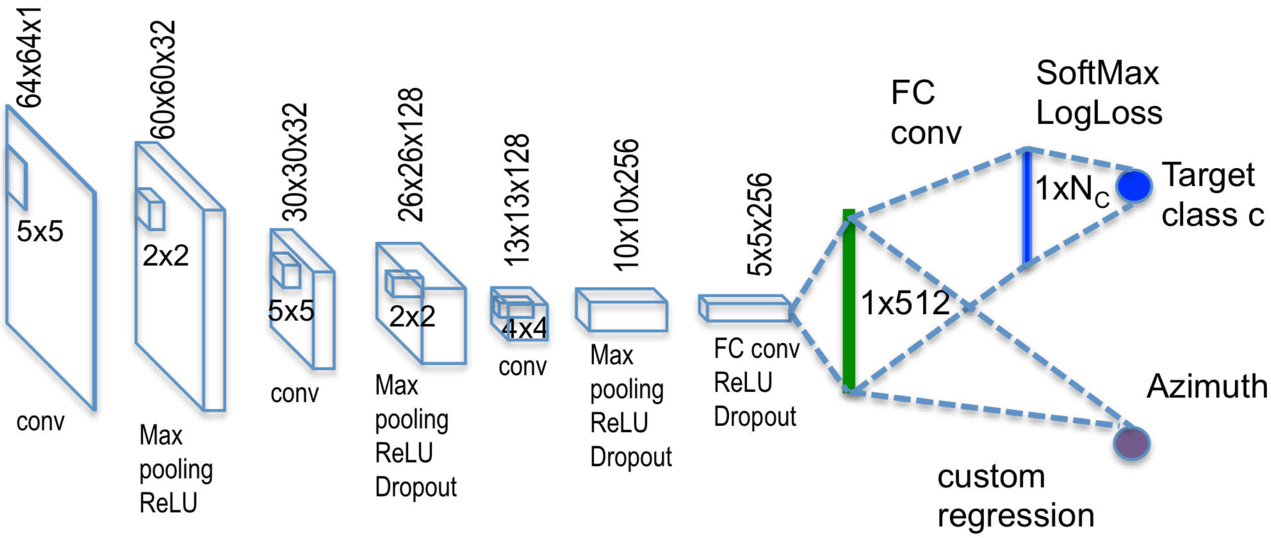


Figure 3. Architecture of the pose-aware DCNN used for SAR ATR experiments on the public MSTAR dataset.

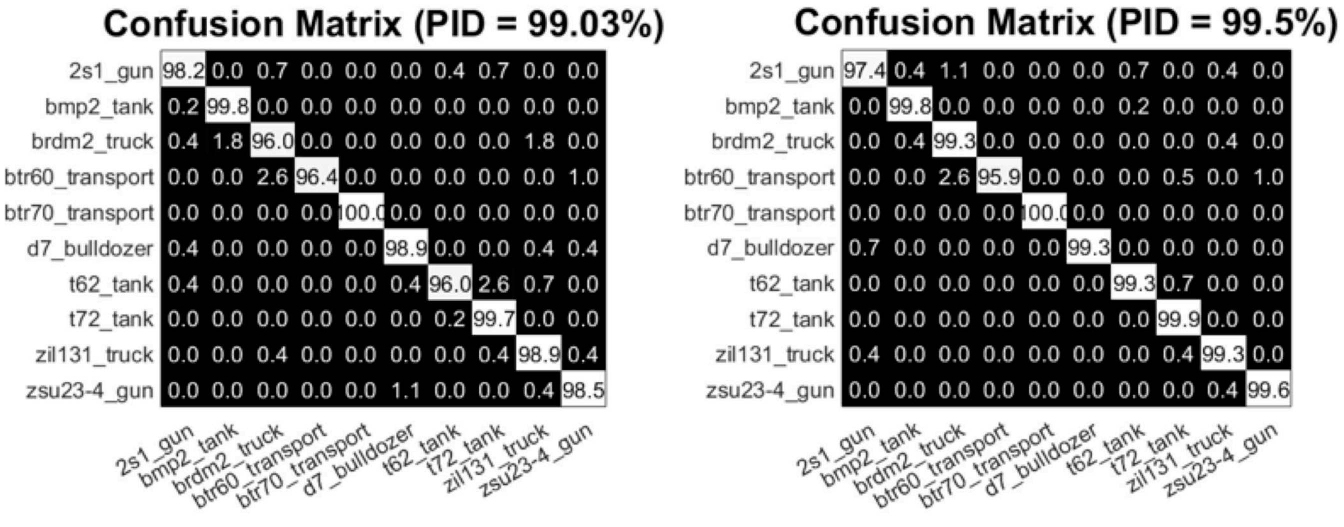
Table 1. Parameter values used in DCNN training

Momentum	0.9
Weight decay	0.0005
Learning rate	0.001
Weight λ for $\boldsymbol{obj}^{\boldsymbol{POSE}}$	1.0

To assess the advantage of the pose-aware DCNN architecture, we also evaluate the performance of a baseline model, which is identical except that the pose constraint is removed during training, so that the performance difference between the two trained deep nets is solely due to the pose reasoning.

We show on the left and right side of Figure 4 the confusion matrices of the baseline network and the proposed pose-aware DCNN respectively, using test set of the MSTAR dataset. Even though our baseline architecture has performed very well with an accuracy of 99.03% over all test images, the auxiliary pose information in training has sculptured the pose-aware DCNN to achieve an overall accuracy of 99.50%, which is almost an 50% reduction in relative error.

Table 2 compares the performances of our proposed algorithm and existing approaches on the MSTAR dataset, including four top performing traditional ATR algorithms and two deep learning approaches. Not surprisingly, the top three performances are achieved using DCNNs: our baseline model, A-ConvNets [54], and the proposed pose-aware model. Note that our baseline model performs



Confusion Matrix (PID = 99.03%)

	2s1_gun	bmp2_tank	brdm2_truck	btr60_transport	btr70_transport	d7_bulldozer	t62_tank	t72_tank	zil131_truck	zsu23-4_gun
2s1_gun	98.2	0.0	0.7	0.0	0.0	0.0	0.4	0.7	0.0	0.0
bmp2_tank	0.2	99.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
brdm2_truck	0.4	1.8	96.0	0.0	0.0	0.0	0.0	0.0	1.8	0.0
btr60_transport	0.0	0.0	2.6	96.4	0.0	0.0	0.0	0.0	0.0	1.0
btr70_transport	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0
d7_bulldozer	0.4	0.0	0.0	0.0	0.0	98.9	0.0	0.0	0.4	0.4
t62_tank	0.4	0.0	0.0	0.0	0.0	0.4	96.0	2.6	0.7	0.0
t72_tank	0.0	0.0	0.0	0.0	0.0	0.0	0.2	99.7	0.0	0.0
zil131_truck	0.0	0.0	0.4	0.0	0.0	0.0	0.0	0.4	98.9	0.4
zsu23-4_gun	0.0	0.0	0.0	0.0	0.0	1.1	0.0	0.0	0.4	98.5

Confusion Matrix (PID = 99.5%)

	2s1_gun	bmp2_tank	brdm2_truck	btr60_transport	btr70_transport	d7_bulldozer	t62_tank	t72_tank	zil131_truck	zsu23-4_gun
2s1_gun	97.4	0.4	1.1	0.0	0.0	0.0	0.7	0.0	0.4	0.0
bmp2_tank	0.0	99.8	0.0	0.0	0.0	0.0	0.2	0.0	0.0	0.0
brdm2_truck	0.0	0.4	99.3	0.0	0.0	0.0	0.0	0.0	0.4	0.0
btr60_transport	0.0	0.0	2.6	95.9	0.0	0.0	0.0	0.5	0.0	1.0
btr70_transport	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0
d7_bulldozer	0.7	0.0	0.0	0.0	0.0	99.3	0.0	0.0	0.0	0.0
t62_tank	0.0	0.0	0.0	0.0	0.0	0.0	99.3	0.7	0.0	0.0
t72_tank	0.0	0.0	0.0	0.0	0.0	0.0	0.0	99.9	0.0	0.0
zil131_truck	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.4	99.3	0.0
zsu23-4_gun	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.4	99.6

Figure 4. Confusion matrix (in percent) for SAR target classification using pose-aware DCNN (right) and baseline DCNN without pose information (left) on the public MSTAR dataset. Rows: truth; columns: result.

statistically tied to A-ConvNets, 99.03% versus 99.1% in classification accuracy, even though we have used an augmented training set that is less than half of that used to train A-ConvNets: our augmented set of 12,146 training images using flip-augmentation, versus the training set of near 27,000 images using position jittering for A-ConvNets. The proposed pose-aware DCNN improves upon the conventional architecture and achieves the best overall classification accuracy of 99.50% among currently published algorithms for the MSTAR dataset.

Table 2. SAR ATR performance comparison with published algorithms.

Algorithm	Accuracy/St. Dev. (%)
Bayesian Compressive Sensing [60]	92.6/NA
SRMS [10]	93.6/NA
Conditionally Gaussian Model [38]	96.9/NA
Modified Polar Mapping [39]	98.8/NA
Basic DCNN [36]	92.3/NA
A-ConvNets [54]	99.1/NA
Proposed baseline DCNN	99.03/0.13
Proposed Pose-aware DCNN	99.50/0.10

6. Conclusions

To take full advantage of deep neural networks we need to better understand how they work to solve highly complex real world challenges. Taking clues from observations that deep networks capture attributes or functionalities that do not directly associate with the tasks they are trained on, we perceive that deep networks build holistic and general representations in order to optimize an objective on a dataset full of variations from many different sources. Recognizing that deep nets perform unsupervised learning of impacting latent factors during the supervised learning of a specific objective, we propose to boost training with available information on the auxiliary explanatory factors to obtain networks with better comprehension of the data population. We describe a general framework to incorporate knowledge of explanatory factors into the deep model for improved performance. We demonstrate the merit of the framework in improving performance of standard DCNNs in the application of pose-aware object recognition.

As the world is full of variations and confounding factors are omnipresent for practical problems, our findings open up new possibilities to improve the performances of DCNNs. For example, it is possible to improve face identification and verification by augmenting the training procedure with information on confounding factors such as pose, lighting condition, facial expression, age, gender, etc. We will explore applying this principle to additional deep learning applications with different explanatory factors.

Acknowledgement

This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA). The views, opinions and/or findings expressed are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government. Approved for Public Release, Distribution Unlimited.

References

[1] F. Anselmi, L. Rosasco, C. Tan, & T. Poggio, Deep convolutional networks are hierarchical kernel machines. *arXiv preprint arXiv:1508.01084*, 2015.

[2] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new *Intelligence, IEEE Transactions on* 35.8 (2013): T. L. Berg, A. C. Berg, and J. Shih. Automatic attribute In L. Bourdev, S. Maji, and J. Malik. Describing people: Poselet-based attribute classification. In *Proc. ICCV*, 2011.

[3] R. Caruana, Multitask learning. *Machine learning* 28.1, 1997.

[4] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems, 2016.*

[5] B. Cheung, J.A. Livezey, A.K. Bansal, and B.A. Olshausen. Discovering hidden factors of variation in deep networks. *CoRR*, abs/1412.6583, 2014.

[6] R. Collobert, and J. Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. *Proc. 25th ICML*, ACM, 2008.

[7] G. Dahl, T.N. Sainath, and G. E. Hinton. Improving deep neural networks for LVCSR using rectified linear units and dropout. *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013.

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

[9] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv preprint arXiv:1310.1531*, 2013.

[10] G. Dong, N. Wang and G. Kuang, Sparse representation of monogenic signal: With application to target recognition in SAR images, *IEEE Signal Processing Letter*, vol. 21, no. 8, pp. 952-956, 2014.

[11] D. Eigen, and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2650-2658. 2015.

[12] D. Erhan, Y. Bengio, A. Courville, P.A. Manzagol, P. Vincent, & S. Bengio, (2010). Why does unsupervised pre-training help deep learning? *The Journal of Machine Learning Research*, *11*, 625-660.

[13] G.J. Ettinger, G.A. Klanderman, W.M. Wells III, W.E. Grimson. Probabilistic optimization approach to SAR feature matching. In Aerospace/Defense Sensing and Controls, pp. 318-329. International Society for Optics and Photonics, 1996.

[14] J. Fan, T. Zhao, Z. Kuang, Y. Zheng, J. Zhang, J. Yu, and J. Peng, HD-MTL: Hierarchical Deep Multi-Task Learning for Large-Scale Visual Recognition. IEEE Trans. Image Processing 26(4): 1923-1938 (2017).

[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proc. CVPR*, 2014.

[16] I.J. Goodfellow, D. Warde-Farley, M. Mirza, A.C. Courville, and Y. Bengio, 2013. Maxout networks. *ICML (3)*, *28*, pp.1319-1327.

[17] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *Computer vision–ECCV 2014*, pp. 297-312. Springer International Publishing, 2014.

[18] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *arXiv preprint arXiv:1502.01852*, 2015.

[19] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. CVPR, 2016

[20] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*, abs/1207.0580, 2012.

[21] J. Hoffman, S. Guadarrama, E. S. Tzeng, R. Hu, J. Donahue, R. Girshick, T. Darrell, and K. Saenko. LSDA: Large scale detection through adaptation. In *Advances in Neural Information Processing Systems*, pp. 3536-3544. 2014.

[22] S. Hong, H. Noh, B. Han, Decoupled Deep Neural Network for Semi-supervised Semantic Segmentation, *NIPS 2015.*

[23] B.K.P. Horn, Closed form solution of absolute orientation using unit quaternions. J. Opt. Soc. Am. **4**(4), 629–642 1987.

[24] J. Hu, J. Lu, and Y.-P. Tan. Discriminative deep metric learning for face verification in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1875– 1882, 2014.

[25] G. Huang, Z. Liu, L. van der Maaten, & K. Q. Weinberger, Densely connected convolutional networks, CVPR 2017.

[26] D.Q. Huynh, Metrics for 3D rotations: Comparison and analysis, Journal of Mathematical Imaging and Vision, (35):2, pp. 155-164, 2009.

[27] S. Ioffe and C. Szegedy Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, arXiv:1502.03167, v3, 2015.

[28] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM International Conference on Multimedia*, pages 675–678, 2014.

[29] P. Khorrami, T. Paine, and T. Huang. Do Deep Neural Networks Learn Facial Action Units When Doing Expression Recognition?. *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 2015.

[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[31] Z. Kuang, Z. Li, T. Zhao, J. Fan, Deep Multi-task Learning for Large-Scale Image Classification. BigMM 2017: 310-317.

[32] T.D. Kulkarni, W.F. Whitney, P. Kohli, and J. Tenenbaum. Deep convolutional inverse graphics network. In *Advances in Neural Information Processing Systems*, 2015.

[33] Y. LeCun, Y., L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324, 1998.

[34] A. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier nonlinearities improve neural network acoustic models. *Proc. ICML*. Vol. 30. 2013.

[35] M. Mathieu, J. Zhao, P. Sprechmann, A. Ramesh, and Y. LeCun. Disentangling factors of variation in deep representations using adversarial training. *arXiv preprint arXiv:1611.03383*.

[36] D. Morgan. Deep convolutional neural networks for ATR from SAR imagery. *SPIE Defense+ Security*. International Society for Optics and Photonics, 2015.

[37] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Weakly supervised object recognition with convolutional neural networks, 2014.

[38] J. O'Sullivan, M. DeVore, V. Kedia, and M. Miller, Sar atr performance using a conditionally gaussian model, Aerospace and Electronic Systems, IEEE Transactions on 37, 91–108, Jan 2001.

[39] J. I. Park and K. T. Kim, Modified polar mapping classifier for SAR automatic target recognition, *IEEE Trans. on Aerospace and Electronic Systems*, vol. 50, pp. 1092-1107, 2014.

[40] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Image understanding* 3.1-18 :111, 1989.

[41] T.D. Ross, J.J. Bradley, L.J. Hudson, and M.P. O'Connor, SAR ATR – so what's the problem? An MSTAR perspective. In E.G. Zelnio (Ed.), Algorithms for Synthetic Aperture Radar Imagery VI (Proceedings of SPIE), 3721, 1999.

[42] T.D. Ross, S.W. Worrell, V.J. Velten, J.C. Mossing, and M.J. Bryant. Standard SAR ATR evaluation experiments using the mstar public release data set, Proc. SPIE 3370, 566–573, 1998.

[43] P. Samangouei and R. Chellappa. Convolutional Neural Networks for Facial Attribute-based Active Authentication on Mobile Devices. *arXiv:1604.08865* (2016).

[44] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *International Conference on Learning Representations (ICLR 2014)*.

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

[46] R. K. Srivastava, K. Greff, and J. Schmidhuber. Highway networks. *arXiv preprint arXiv:1505.00387* (2015).

[47] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller. Multi-view Convolutional Neural Networks for 3D Shape Recognition, ICCV 2016.

[48] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In

Advances in Neural Information Processing Systems, pages 1988–1996, 2014.

[49] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. *arXiv preprint arXiv:1412.1265*, 2014.

[50] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*, 2014.

[51] Y. Taigman, M. Yang, M. A. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1708, 2014.

[52] A. Vedaldi, and K. Lenc. MatConvNet: Convolutional neural networks for matlab. *Proc. 23rd Annual ACM Conference on Multimedia Conference*. ACM, 2015.

[53] S. Wager, S. Wang, and P.S. Liang. Dropout training as adaptive regularization. *Advances in neural information processing systems*. 2013.

[54] H. Wang, S. Chen, F. Xu, and Y. Jin, Application of deep-learning algorithms to MSTAR data, IEEE Int'l Conf. Geoscience and Remote Sensing Symposium (IGARSS), 2015.

[55] D. Yamins, H. Hong, C. Cadieu, E. Solomon, D. Seibert, and J. DiCarlo, Performance-optimized hierarchical models predict neural responses in higher visual cortex, *Proceedings of the National Academy of Sciences*, 2014.

[56] J. Yang, S.E. Reed, M. Yang, and H. Lee. Weakly-supervised disentangling with recurrent transformations for 3d view synthesis. In *Advances in Neural Information Processing Systems*, 2015.

[57] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks?. In *Advances in Neural Information Processing Systems*, pp. 3320-3328. 2014.

[58] M.D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. *Computer vision–ECCV 2014*. Springer International Publishing, 2014. 818-833.

[59] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev. PANDA: Pose aligned networks for deep attribute modeling. In *Proc. CVPR*, 2014.

[60] X. Z. Zhang, J. H. Qin and G. J. Li, SAR Target Classification Using Bayesian Compressive Sensing with Scattering Centers Features, *Progress in Electromagnetics Research-Pier*, vol. 136, pp. 385-407, 2013.

[61] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Facial landmark detection by deep multi-task learning. In *Computer Vision–ECCV 2014*, pp. 94-108. Springer International Publishing, 2014.

[62] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object Detectors Emerge in Deep Scene CNNs. *International Conference on Learning Representations (ICLR), 2015.*