What's Hiding in my Deep Features?

Ethan M. Rudd[1], Manuel Günther[1], Akshay R. Dhamija[1], Faris A. Kateb[1], and Terrance E. Boult[1]

[1]Vision and Security Technology Lab, University of Colorado Colorado Springs

Abstract

Input variations are often irrelevant to the desired target output of a recognition system, for example, face images of the same person that differ in pose or facial expression should be ascribed the same identity. Deep neural networks are often hailed for their ability to learn representations that are invariant across a wide range of input variations. This invariance is often assumed from overall performance, but research has demonstrated the contrary, e.g., that deep neural networks are sensitive to out-of-plane rotations. Simultaneously, some approaches implicitly rely on non-invariant properties, for instance, using a truncated network trained on facial identities to arrive at a representation used to classify facial attributes. In this work, we study non-invariant properties of large face recognition networks in detail, demonstrating that networks trained on face identities work quite well for classifying not only identity-related attributes, but can also effectively classify attributes that are only slightly correlated or uncorrelated with face identity. Facial expression or accessory related attributes, for example, are attributes that we would expect a truly invariant identity-trained network to attenuate or ignore. However, noticeable information about facial expression and accessories is still contained within the penultimate layer of the network, as is noticeable information regarding pitch, roll, and yaw. Non-invariant properties of a feature space need not preclude good recognition performance in an end-to-end network, provided that classes effectively separate/saturate at the final output. Using a non-invariant feature space derived from a truncated network can also allow generalization to novel tasks, e.g., attribute prediction or face verification from a face-identity trained network. However, there are some situations in which an invariant representation is desirable, for example, when enrolling new identities in a recognition system. If the feature space is non-invariant, just because identities in the training set were able to separate/saturate well, there is no guarantee that newly enrolled samples or identities will have a similar property. To this end, and on a more theoretical note, we induce variations on the MNIST dataset and augment the LeNet architecture by adding several invariance constraints on the feature space in the objective function. Analysis suggests that while these steps do lead to certain invariant characteristics, arriving at a non-invariant feature space may be difficult to accomplish.

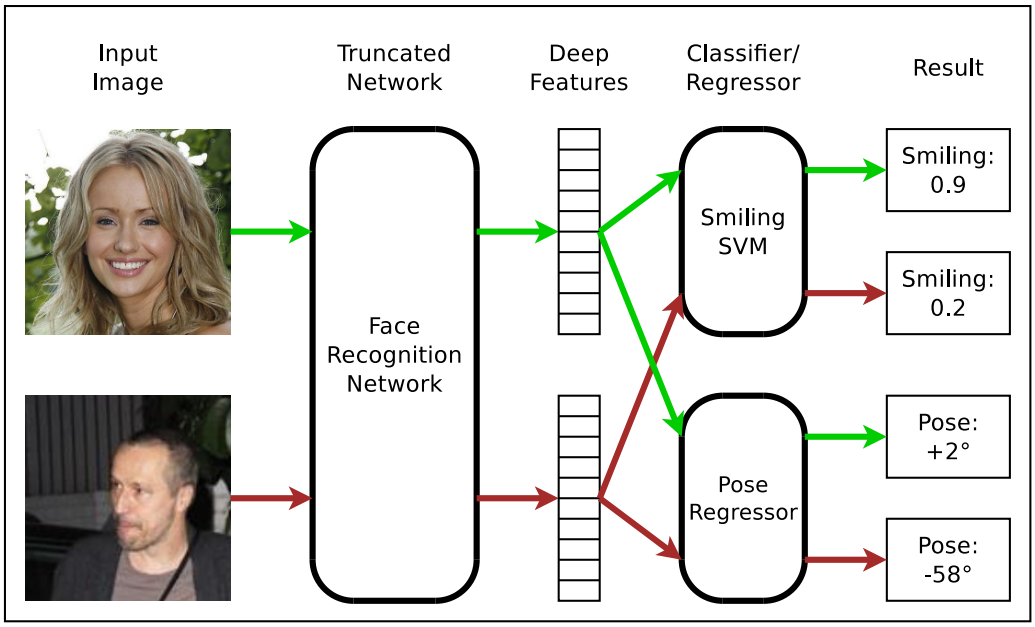


Figure 1: Classifying Image Properties from Deep Features. *In a good face recognition system, non-identity-related properties such as smiling or pose should not play a role. So, people expect that deep features extracted from truncated face recognition networks are invariant to these properties. But they are not. We show that using simple classifiers or regressors, these image properties can be predicted with high reliability from the deep features.*

1 Introduction

Effective recognition hinges on the ability to map inputs to their respective class labels despite exogenous input variations. For example, the associated identity label from a face recognition system should not change due to variations in pose, illumination, expression, occlusion, or accessories. Deep neural networks have been hailed for their ability to deliver state-of-the-art recognition performance, even under noticeable input variations. While raw outputs from end-to-end networks are commonly used in idealized benchmark settings, e.g., on the ImageNet challenge [18], realistic tasks seldom use actual network outputs because training a full network is time consuming and classes in the application at hand are often not so rigidly defined as in benchmark settings. Moreover, fusing additional information through backpropagation is sometimes non-trivial.

Particularly in biometrics applications, a network that is trained on many examples for a specific modality is commonly used as a feature extractor, i.e., by taking feature representations from lower layers of the network. During enrollment, samples from the gallery are submitted to the network and the output feature vectors are used for template construction. These templates may be constructed in a variety of ways, e.g., as sets of

raw feature vectors or aggregations thereof (e.g., [15, 2]). At match time, probe samples are compared to the gallery, e.g., by taking the cosine distance to gallery templates [2]. Critically, operational constraints often require fast enrollment, which precludes training an end-to-end network and significantly reduces the likelihood that samples in the gallery will be from the same classes as samples in the training set – many face recognition protocols even require that identities in training and enrollment sets have no overlap.

Unlike the outputs of an end-to-end network, in which learnt classifications exhibit invariances to exogenous factors in the input (they have to, assuming reasonable recognition accuracy), "there is no concrete reasoning provided [in the literature] on the invariance properties of the representations out of the fully connected layers" [16], nor is there any intuitive reason to assume that deep features prior to the final fully-connected layer will exhibit such invariances because typical loss functions have no explicit constraint to enforce such invariance. While some exogenous properties of input samples are presumably attenuated by the network, since these higher-level abstractions still offer quite good recognition performance, there is still significant motivation to discern what type of exogenous information resides in these features.

As a primary motivation, note that lack of invariance is not necessarily a bad thing: Exogenous information preserved from the input is partially what allows deep representations to generalize well to other tasks. For example, popular approaches to facial attribute classification use features derived from networks trained on identification tasks [13]. A truly invariant representation would preclude information about attributes that are unrelated to identity from being learnt in such a manner (e.g., `Smiling`), yet until recently, this was the state-of-the-art facial attribute classification approach and is still widely used. The new state of the art [17, 5] leverages attribute data directly, precisely for this reason. More generally, many learning approaches to task-transfer and domain adaptation often make the implicit assumption that a feature space derived from a different task or a different domain will work well for the task/domain at hand, provided that enough data has been used to derive the feature space, with the actual transfer learning conducted via classifiers within this feature space. We show in this work that such an assumption may or may not be true depending on the information content of the feature space.

A secondary motivation for understanding the nature of exogenous information embedded in deep features – namely, the desire to create more invariant representations – becomes important when a truncated network is used to enroll novel samples not present in the training set; specifically if those samples constitute novel identities. The original end-to-end network is optimized so that classes constituted by the original training samples separate well – even if the underlying deep features are not invariant to exogenous factors, the last layer will combine these deep features into an overall classification. Since we ignore the last layer during enrollment, variations due to exogenous factors could result in confusion between classes for newly enrolled samples. Thus, there is reason to explore whether we can attain more invariant deep feature spaces since this could increase performance scalability of applied machine learning systems – especially for biometric recognition.

In this work, we conduct a formal exploration of the invariant and non-invariant properties of deep feature spaces. While there has been related research conducted in the machine learning and computer vision communities in areas of domain adaptation and task transfer learning, there has been little direct concentration on this topic, in part we surmise, due to the already impressive recognition rates and representational performance that deep neural networks provide. We analyze two deep representations. The first, geared toward realistic applications, is the combined output of multiple face recognition networks [1, 19] – an approach which achieved state-of-the-art accuracy on the IJB-A dataset [9]. As indicated in Figure 1, we demonstrate that from this representation we can not only accurately predict pose, we can also predict facial attributes, despite an added triplet-loss metric learning phase atop these representations. With respect to attribute classification, we find that classifiers trained in this deep feature space – which is purely identity-derived – achieve near state-of-the-art performance on identity related attributes such as gender and, surprisingly, they are also able to achieve impressively high performance on non-identity related attributes (e.g., `Smiling`), which an invariant representation would have down-weighted or pooled out. The second deep representation that we employ is the canonical LeNet/MNIST architecture [11], with which we attempt several different training procedures to enforce invariant representations. Our analysis demonstrates that we are indeed able to extract a more invariant feature space with little accuracy loss, but many non-invariances still remain.

2 Related Work

Face biometric systems have seen remarkable performance improvements across many tasks since the advent of deep convolutional neural networks. Taigman *et al.* [20] pioneered the application of modern deep convolutional neural networks to face recognition tasks, with the first network (DeepFace) to reach near human verification performance on the Labeled Faces in the Wild (LFW) benchmark [7]. In their work, they used an external image preprocessing to frontalize images, and trained their network on a private dataset of 4.4 million images of more than 4000 identities. Later, Oxford's Visual Geometry Group (VGG) publicly released a face recognition network [15] that omits the frontalization step, while training the network with a relatively small dataset containing 95 % frontal and 5 % profile faces. Parkhi *et al.* [15] also implemented a triplet-loss embedding and demonstrated comparable performance to [20] on LFW despite the lower amount of training data. Lately, the IJB-A dataset and challenge [9] was proposed, where more profile faces are included. Chen *et al.* [2] trained two networks on a small-scale private dataset, containing more profile faces than the DeepFace and VGG training sets. Using a combination of these two networks and a triplet-loss embedding that was optimized for comparing features with the dot product, they reached the current state-of-the-art results on the IJB-A challenge. The combination of these deep features is the basis for our analysis in Section 3.

Facial attribute classification using deep neural network was pioneered by Liu *et al.* [13]. The authors collected a large-scale dataset (CelebA) of more than 200,000 images, which they labeled with 40 different facial attributes. They trained a series of two localization networks (LNets) and one attribute classification network (ANet). The ANet was pre-trained on a face identification task, and fine-tuned using the training partition of CelebA attribute data. Finally, they trained individual support vector machines (SVMs) atop the learnt deep features of the penultimate layer of their ANet to perform final attribute prediction. Wang *et al.* [22] pre-trained a network using data that they collected themselves via ego-centric vision cameras and augmented that dataset with ground-truth weather and geo-location information. They then fine-tuned it on the CelebA training set. While previous approaches that advanced the state of the art on CelebA relied on augmented training datasets and separate classifiers trained atop end-to-end deep features, Rudd *et al.* [17] recently advanced the state of the art beyond these using an end-to-end network trained only on the CelebA training set, but with a multi-task objective, optimizing with respect to all attributes simultaneously. Their mixed objective optimization network (MOON) is based on the VGG topology, and also introduces a balancing technique to compensate for the high bias for some attributes in CelebA. Finally, Günther *et al.* [5] extended the approach of [17] to accommodate unaligned input face images using the alignment free facial attribute classification technique (AFFACT), which is able to classify facial attributes using only the detected bounding boxes, i.e., without alignment. This network provides the current state of the art on the CelebA benchmark using no ground truth landmark locations from the test images. In this book chapter, we investigate a network that is a clone of the AFFACT network, which was trained using the same balancing method presented in [17].

The use of deep learnt representations across visual tasks, including the aforementioned face biometric, can be traced back to the seminal work of Donahue *et al.* [3], in which the authors used a truncated version of AlexNet [10] to arrive at a deep convolutional activation feature (DeCAF) for generic visual recognition. Several efforts to remove/adapt variations in features that transfer across domains or tasks have been conducted, including some that are similar in nature to our research [12, 16, 21]. Li *et al.* [12] introduced a multi-scale algorithm that pools across domains in an attempt to achieve invariance to out-of-plane rotations for object recognition tasks. Mopuri and Babu [16] formulated a compact image descriptor for semantic search and content-based image retrieval applications with the aim of achieving scale, rotation, and object placement invariance by pooling deep features from object proposals. Tzeng *et al.* [21] formulated a semi-supervised domain transfer approach that, during fine-tuning to the target domain, uses a new loss-function that combines standard softmax loss with a "soft label" distillation (cf. [6]) to the softmax output mean vector and a domain confusion loss for domain alignment, which iteratively aims to first optimize a classifier that best separates domains then optimize the representation to degrade this classifier's performance. The approach in [21] is very similar to one of our methods in Section 4, but the goal is different – their approach aims to transfer domains within and end-to-end deep network, while ours aims to obtain an invariant representation

for training lighter-weight classifiers on new samples from an unspecified target domain.

The use of pre-trained deep representations is not new to face biometrics, but investigating the content of these deep features has recently attained interest. Particularly, Parde *et al.* [14] investigated how well properties of deep features can predict non-identity-related image properties. They examined images from the IJB-A dataset [9] and concluded that "DCNN features contain surprisingly accurate information about yaw and pitch of a face." Additionally, they revealed that it is not possible to determine individual elements of the deep feature vector that contained the pose information, but that pose is encoded in a different set of deep features for each identity. Our work builds on theirs by investigating pose and attribute-related information content across identity-trained deep representations.

3 Analysis of a Face Network

A typical example of a face image processing network is a face recognition network [15, 2]. These networks are usually trained on large datasets using millions of images of thousands of identities, with the objective of minimizing negative log likelihood error under a softmax hypothesis function. Choice of training set is often made to capture wide variations both within and between identities, so training sets usually contain images with a wide variety of image resolutions, facial expressions, occlusions, and face poses. The resulting network is generally able to classify the correct training identities independently of the presence of these traits in the images.

Contrary to many closed-set image classification tasks, one characteristic of face recognition systems in practice is that the identities in the training and test sets differ. As previously mentioned, this stems predominantly from the fact that training a deep neural network is computationally expensive, but in deployment settings novel identities must be enrolled frequently. Hence, the softmax output for a given test image provides little value since it saturates with respect to training identities. While one could train a secondary classifier atop softmax outputs, this is typically not done because the saturating effects remove potentially useful information. Instead, common practice is to use the output of the pre-softmax layer of the network as a feature vector to represent an input face image. We will refer to vectors within this vector space as *deep features*.

To compute the similarity between two faces, the deep features are compared using some sort of distance function or classifier, e.g., Euclidean distance [15], cosine distance [2], and Siamese networks [20]. To increase classification accuracy, a metric-learnt embedding is added, e.g., triplet loss [15, 19] or joint Bayesian [20], which projects the deep features into a space that is trained to increase the similarity of deep features from the same identity, while decreasing the similarity of deep features extracted from different identities. Due to the enormous boost in face recognition performance that these sorts of systems have provided, a prevailing, albeit disputed and factually ungrounded belief is that the deep features are mostly independent of image parameters that are not required for face recognition. In this

section, however, we show that we are able to classify non-identity-related facial attributes as well as face pose from the deep features – both before and after triplet loss embedding, demonstrating that deep features maintain significant information in their representation about image parameters that are independent from identity.

3.1 Attribute Prediction Experiments

To investigate the independence of deep features from non-identity-related facial attributes, we performed experiments on the CelebA dataset [13], which contains around 200,000 face images, each of which is hand-annotated with 40 binary facial attributes. We obtained the deep features (832-dimensional before and 256-dimensional after triplet loss embedding) and the according hyperface annotations from Chen *et al.* [2] for all images of the CelebA dataset. Using the deep features from the CelebA training set, we trained 40 linear support vector machines (SVMs) – one for each attribute – and optimized the C parameter for each attribute individually using the CelebA validation set. Due to the large imbalance of many of the attributes within the CelebA dataset (cf. [17]), e.g., approximately 98% of the images lack the `Bald` attribute, we trained all SVM classifiers to automatically balance between positive and negative labels. On the test set, we predicted all 40 attributes and compared them with the ground-truth labels from the dataset and split the errors into false positives and false negatives. False positives in this case corresponds to an attribute labeled as absent, but predicted to be present.

In order to get an idea, how well the attributes can be predicted, we trained a version of the AFFACT network [5] using the attribute balancing proposed in [17]. The results of this experiment are shown in Figure 2, where we have split the 40 attributes into identity-related, -correlated and -independent. For identity-dependent attributes such as `Male`, `Pointy Nose`, and `Bald` we can see that the prediction from the deep features results in approximately the same error as AFFACT's. Hence, these attributes are well-contained in the deep features, despite the fact that the network was not trained to predict these attributes. Exceptions are attributes like `Narrow Eyes` and `Oval Face`. These attributes have a high overall prediction error and we hypothesize that they may have been difficult for dataset providers to consistently label.

Identity-correlated attributes, e.g., hair style and hair color, are usually stable but might change more or less frequently. These attributes are generally more difficult to predict from the deep features. While predictions from the deep features are still better than random, the associated errors are considerably higher than those corresponding to classifications made by the AFFACT network. Note that the features after the triplet-loss embedding predict the identity-correlated attributes worse than before the embedding. Interestingly, for some attributes like `Gray Hair` or `Sideburns` we can observe a difference in false positives and false negatives. While we can predict the *absence* (low false positives) of these attributes similarly or even better than AFFACT, the prediction of the *presence* (low false negatives) of them is reduced for the deep features.

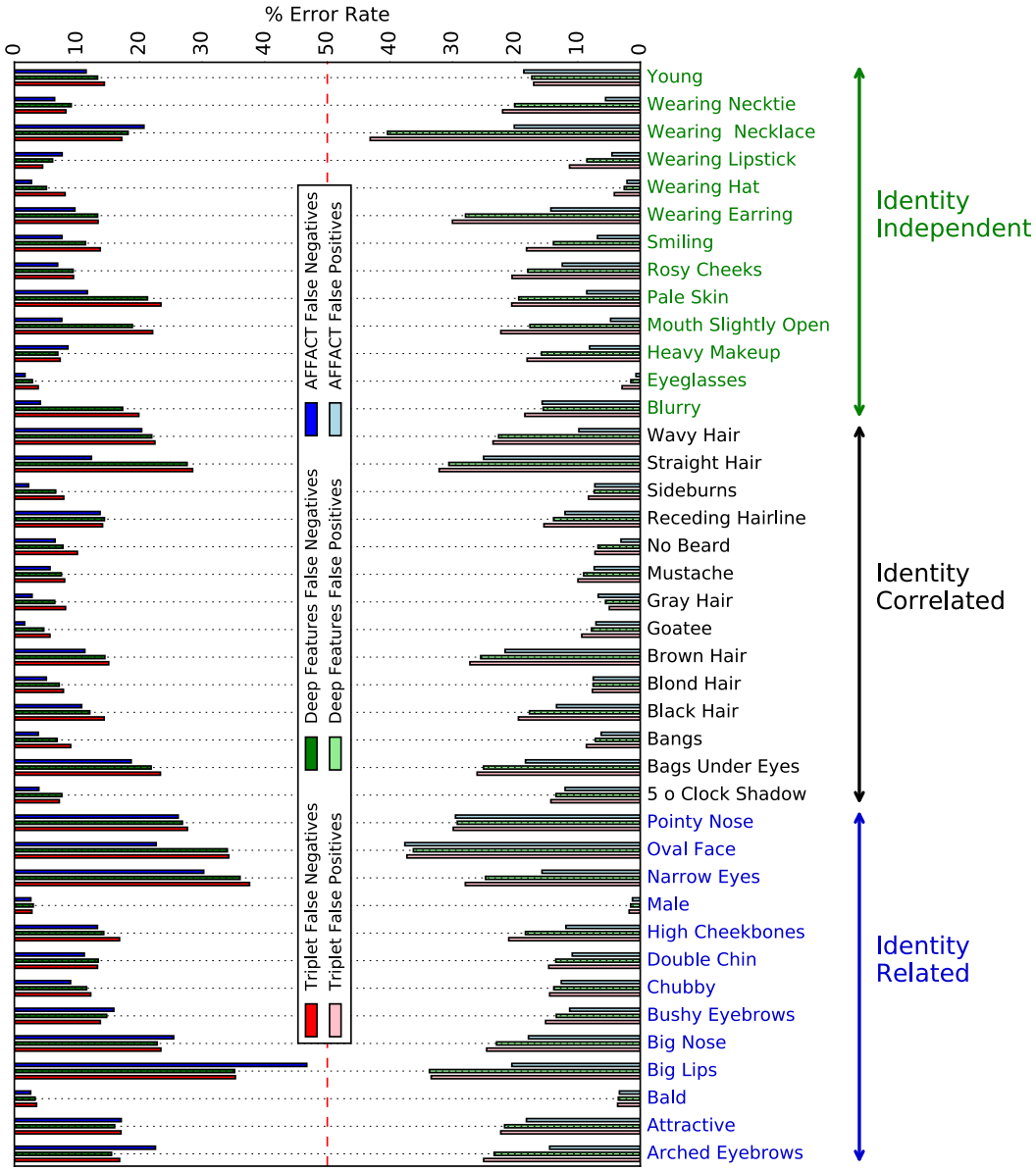


Figure 2: Attribute Prediction from Deep Features. *The error in attribute prediction from the deep features before and after triplet loss embedding is displayed, split into false negatives (attributes labeled as present but predicted as absent) on the left and false positives (absent attributes predicted as present) on the right. For comparison, the AFFACT network results show the state-of-the-art attribute prediction. Attributes are loosely grouped into identity-dependent, identity-correlated, and identity-independent subsets.*

Finally, the identity-independent attributes such as `Smiling`, `Blurry`, and `Wearing Necklace` are far more difficult to predict from the deep features than from the AFFACT network, but the classification is still noticeably better than random. This suggests that identity-unrelated attribute information is still contained within the deep features, since otherwise both the false positive and the false negative error rates should be 50 %. Hence, the prediction capability of deep features for those attributes is reduced when training the network to classify identity, and even more reduced after triplet-loss embedding, but some attributes like `Wearing Hat` or `Eyeglasses` can still be predicted with very high accuracy. This ultimately means that while some non-identity-related information corresponding to some attributes is attenuated during network training and triplet-loss embedding, other non-identity-related information is preserved, and we do not arrive at a feature space that is truly independent of non-identity-related attribute information.

3.2 Pose Prediction Experiments

When the pose of a face changes from frontal to full profile, its visual appearance alters dramatically. Two faces of different identities in the same pose are generally more similar than the face of the same identity in different poses. Hence, the network needs to learn a representation that is able to differentiate poses from identities, a task that has been shown to be very difficult for non-network-based algorithms [4]. When training the network using softmax, the last layer can combine different elements of the deep features in order to obtain a representation that is independent of pose. In practice this means that the deep features may very well contain the pose information and it is, thus, possible to predict the pose from the deep features.

We performed another experiment on the CelebA dataset, in which we attempt to predict pose from deep features, using the same splits in training, validation and test sets. Since the network was trained with using horizontally flipped images, there is no way to differentiate between positive and negative yaw angles and, hence, we used the absolute yaw angle as target. As the CelebA dataset does not provide the pose information, we took the yaw angles automatically estimated by the state-of-the-art hyperface algorithm algorithm [2] as target values for the pose prediction. Note that the hyperface yaw angle estimates are relatively precise for close-to-frontal images, but for larger yaw angles $> 45°$, they become unreliable.

In order to determine if pose information is generally contained in deep features – not just the deep features of Chen *et al.* [2] – we extracted the penultimate layer from two more networks: the VGG face network [15] (4096-dimensional, layer `FC7`, post-ReLU) used for identity recognition and the AFFACT network [5] (1000-dimensional) that was trained for facial attribute prediction, see above. Despite the different tasks that the networks are trained for, intuition suggests that all of the networks should have learnt their tasks independently of face pose since changes in pose do not change the target labels. For feature extraction, images of the CelebA dataset were cropped according to the detected

Table 1: Predicting Pose from Deep Features. *Result are given for the pose prediction experiments on the CelebA test set, using yaw angles automatically extracted with the hyperface algorithm as target values. Shown are the average absolute difference between predicted and hyperface yaw angles in degrees. The count of images in the according hyperface yaw range (shown left) are given in the rightmost column.*

Yaw	**AFFACT**	**VGG**	**Deep Feat.**	**Triplet-Loss**	**Count**
0–15	6.1	7.7	8.1	10.6	14255
15–30	4.6	6.3	5.8	7.1	4527
30–45	5.6	7.2	5.7	6.7	872
>45	9.1	11.4	10.3	13.9	268
Total	5.8	7.4	7.5	9.7	19922

hyperface bounding box[1] and scaled to resolution 224×224 (cf. [5]), which happens to be the input resolution for both VGG and AFFACT.

For each type of deep features, we trained a linear regression model using the CelebA training set. Using this model, we predicted the yaw angle contained inside the according deep features and compared it with the hyperface yaw angle. The results given in Table 1 display the average distance between the two values in degree. Interestingly, a global trend is that the yaw angle in half-profile pose (from 15° to 45°) could be predicted with the highest precision, while close-to-frontal pose angles seem to be a little more difficult. This suggests that poses up to 15° do not make a large difference for feature extraction, possibly due to the over-representation of data in this range. Upon closer examination of the deep feature types, the outputs of the penultimate layer of the AFFACT network seem to be least stable to pose variations – potentially due to how the network was trained – while the deep features from Chen *et al.* [2] and VGG [15] have higher prediction errors, which are once more superseded by the triplet-loss-projected versions of [2]. Interestingly, we observe the general trend that deep features with more elements (4096 for VGG and 1000 for AFFACT) contain more yaw information than shorter vectors (832 before and 256 after triplet-loss embedding). Given that the average pose prediction errors are generally below 10°, we can conclude that the yaw angle (and we assume that the same is true for pitch and roll angles) can still be predicted from all kinds of deep features and, hence, the deep features are not invariant to pose. Finally, choosing non-linear basis functions could almost certainly enhance pose-prediction accuracy, but the fact that we are able to do so well without them already demonstrates the presence of noticeable pose information within the deep features of all three networks.

[1] Both the VGG and AFFACT networks have shown to be stable to different scales and rotation angles. Proper alignment of the face is, hence, not necessary for either of the two networks.

4 Toward an Invariant Representation

In this section we explore the problem of formulating an invariant representation. We perform this exploration using perturbations to the canonical handwritten digit classification dataset MNIST [11], augmenting the familiar LeNet topology that is shipped as an example with the Caffe framework [8].

4.1 Preliminary Evaluation on Rotated MNIST

Since we are interested in investigating the deep feature representation, we do not use the final softmax output of the network during evaluation. This is in contrast to common practice for the MNIST dataset. Instead, we train the network using a softmax loss to learn the representation, but then remove the final layer (`softmax` and `ip2`) of the network to investigate the output of the penultimate (`ip1`) layer. This is analogous to using the penultimate representation for enrollment and recognition in a face recognition network.

The task that we aim to accomplish with this set of experiments is to evaluate the invariance of the `ip1` layer to rotations on the inputs and to explore how to enforce such invariance. As a baseline approach, we artificially rotated the MNIST images using 7 different angles in total: $-45°, -30°, -15°, 0°, 15°, 30°, 45°$. We limited our rotations to this range to avoid legitimate confusions of digits induced by rotation (e.g., 9 and 6). Using the default training images, augmented with each rotation, we trained the standard LeNet network with the default solver until convergence was attained on a similarly augmented form of the validation set. Using the softmax output, we obtained a final classification accuracy of 98.98 % on the augmented form of the test set. This result indicates that the network was able to learn to classify rotated images successfully.

Using the learnt representation, we truncate the network after the `ip1` layer's `ReLU` activation function, extracting 500-dimensional feature vectors for all images rotated at all angles. To represent each of the classes, we simply average the `ip1` feature vectors (which we call the Mean Activation Vector, MAV) of all training images of a class – an approach similar to that commonly used in face pipelines [15]. We ran a simple evaluation on the `ip1` features of the test set by computing the cosine similarity to all 10 MAVs; if the correct class had highest similarity we consider the sample to be classified correctly. Using this approach, the total classification accuracy dropped to 89.90 % – a noticeable decline from using softmax classifications. A more detailed comparison between the two models is given in Figure 4.

Momentarily ignoring the decline in accuracy, a more fundamental question arises: has training across rotations led the representation to become invariant to rotation angle – i.e., is there noticeable information regarding the representation embedded within `ip1`? As a first step in answering this question, we train a linear regressor for each label in the training set and attempt to predict rotation angle from the extracted `ip1` features. As the regression target we use the known rotation angle. On the test set, we classified the `ip1` features with

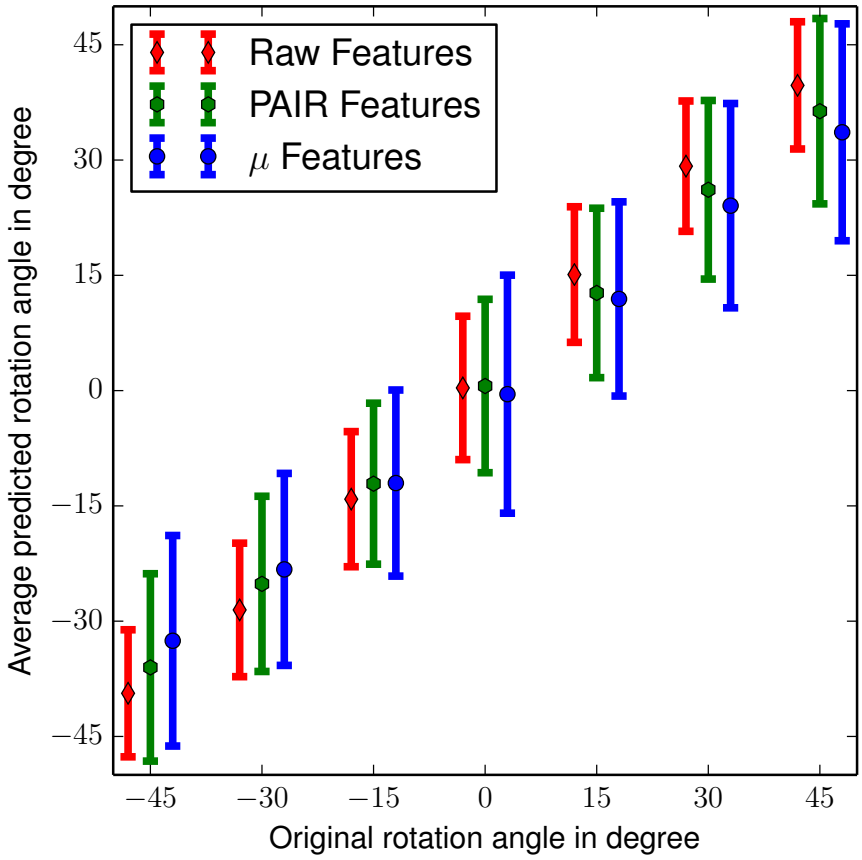


Figure 3: Angle Prediction. *This figure shows the average results of predicting the angles used to rotate the MNIST test images from the* `ip1` *features of LeNet.*

the regressor of the corresponding label. The mean and the standard deviation – averaged over all ten labels – are shown in red color in Figure 3. Even though the original images have an inherent rotation angle – people slant hand-written digits differently – we can reasonably predict the angle with a standard error of around 15°.

Noting our ability to predict pose angle with reasonable accuracy, we then computed an MAV of `ip1` features of each label for each angle separately from the training set. At test time, we estimated the angle of the test `ip1` feature and computed the similarity to all ten MAVs with that angle. Using this approach, our average classification accuracy increased to 95.44 %, cf. Figure 4. Hence, exploiting information from exogenous input features contained in the `ip1` representation actually allows us to improve classification accuracy.

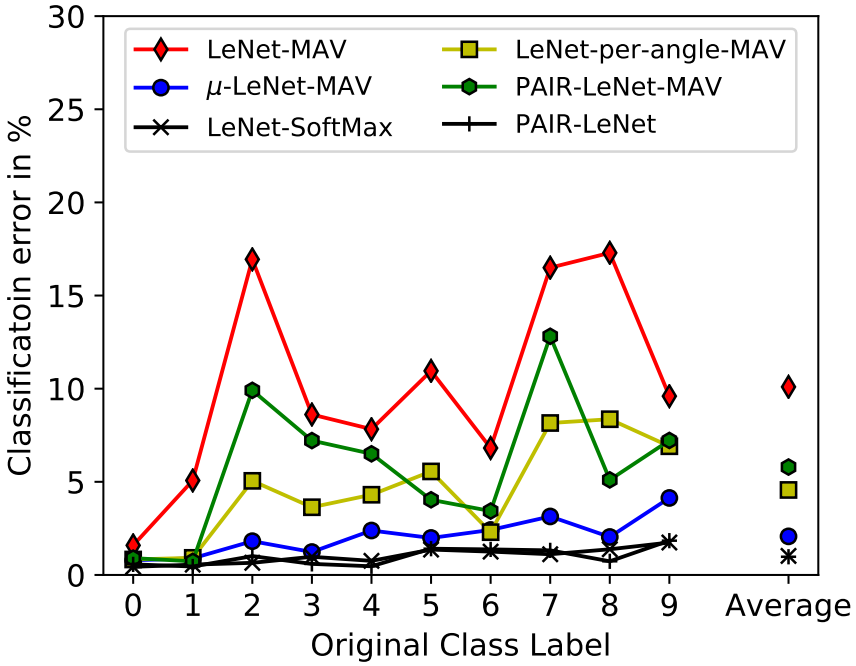


Figure 4: Digit Classification Errors. *Classification errors are given for several techniques, individually for each digit and as an average. Techniques that use the Mean Activation Vector (MAV) are shown in color, while softmax-approaches are given in black and only for comparison. For LeNet-MAV the MAV on the original LeNet is used, evaluation is performed as the class with the lowest cosine distance. For LeNet-per-angle-MAV one MAV per angle is computed on the training set, while the angle is estimated for test images, and evaluation is performed using only the MAVs for this angle. The μ-LeNet network was trained using the corresponding MAV as target, evaluation is similar to LeNet-MAV. The PAIR network was trained using pairs of images, evaluation is similar to LeNet-MAV.*

4.2 Proposed Architectures to Enhance Invariance

In this subsection, we modify LeNet in two distinct ways, for which intuition suggests that it would lead to a representation that is more invariant to exogenous input variations. The first such architecture, μ-LeNet, uses a distillation-like approach by regressing to the mean `ip1` vector across each class at the `ip1` layer of the truncated LeNet. The second architecture, PAIR-LeNet, introduces a Siamese-like topology to encourage two distinct inputs to have the same representation.

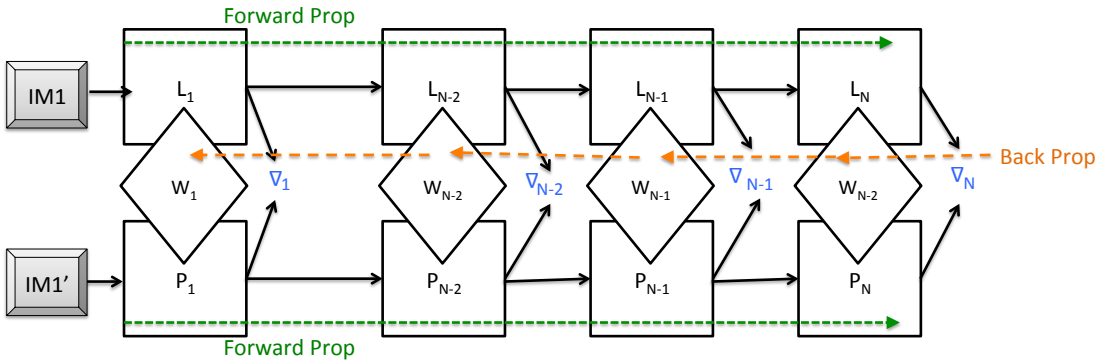


Figure 5: PAIR Network Topology. *A generalized schematic of the Perturbations to Advance Invariant Representations (PAIR) architecture. Pairs of training images are presented to a network with different losses, but shared weights. Conventional softmax loss is used for one of the input images, then per-layer Euclidean losses are used on the differences between the activation vectors of the two images. This aims to force a similar representation between image pairs of the same class.*

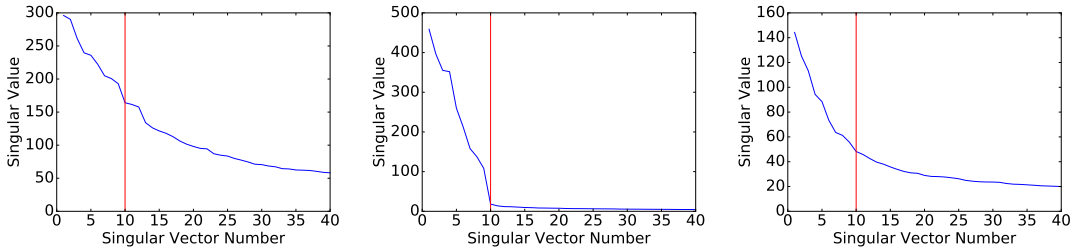
4.2.1 The μ-LeNet

Distillation, introduced by Hinton *et al.* [6] was designed for the purpose of "knowledge transfer" between end-to-end networks, generally of different topologies. The procedure involves softening the output softmax distribution of the network to be distilled, and using the softened distribution outputs as soft labels for the distillation network. Using this as motivation, we use a related but different approach, in which we aim to achieve a more invariant representation. Namely, using the `ip1` output of a trained LeNet, we train a new LeNet `ip1` representation by using the mean activation vector across classes in the original network's `ip1` layer as regression targets. We refer to this topology trained by regressing to the mean activation vector as the μ-LeNet. To stabilize and speed up the learning process, we also performed a min-max normalization on the feature space from 0 to 1 as a preprocessing step.

Using cosine distance with respect to the MAV, our recognition rate was 97.18 %, noticeably better than using the `ip1` layer under the original MNIST topology, cf. Figure 4. While a decrease in angle classification success is noticeable, this decrease was very slight, as shown in blue in Figure 3. The μ-LeNet also took significantly longer time to train than the typical LeNet topology.

4.2.2 The PAIR-LeNet

The μ-LeNet blatantly attempts to force input images to their respective class's mean `ip1` representation output from a trained LeNet. A different approach is to consider pairs of



(a) Raw LeNet (b) μ-LeNet (c) PAIR-LeNet

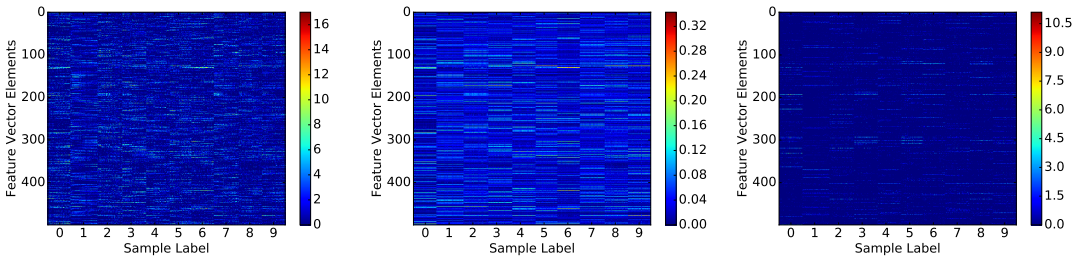
Figure 6: Scree diagrams for derived LeNet/MNIST feature spaces. *These scree plots depict the variance (singular value) explained by the first 40 of 500 singular vectors from the test set data matrices. Red lines reflect the max number of singular vectors expected under an "ideal" representation. All feature vectors were min-max normalized to a* $[-1, 1]$ *range for visual comparison.*

images at a time and optimize their representations to be as identical as possible. We refer to this approach as PAIR – because it uses image pairs with *Perturbations to Advance Invariant Representations*. The idea is depicted in Figure 5, and is similar to a Siamese network topology. We trained the network on the rotated MNIST training set using randomized pairs of rotations until convergence was attained on the validation set. The classification rate of the end-to-end network on the augmented test set was 99.02 %. Using cosine distance with respect to each label and angle mean, we obtained a recognition rate of 94.21 % in the derived `ip1` feature space, cf. Figure 4. As shown in green in Figure 3, angle classification success decreased, but this decrease was only slight. This suggests that, while the topology learns a representation that is useful for recognition, information about pose still resides in the representation.

4.3 Analysis and Visualization of the Proposed Representations

Our MNIST experiments in the previous section demonstrate that we are able to obtain marginal improvements with respect to rotation invariance, but contrary to our expectations, we were suprised how small the improvements are. In this section, we empirically analyze the feature spaces learnt by our proposed approaches to better understand their properties.

Consider an idealized invariant representation designed to characterize the 10 digit classes from MNIST. One characteristic that we would expect the feature space representation to have is a rank no greater than 10. Thus, after subtracting the mean feature vector of the data matrix and performing singular value decomposition (SVD), the vast majority of the variance should be explained by the first singular vectors with the rest accounting for only minor noise.



(a) Raw-LeNet (b) μ-LeNet (c) PAIR LeNet

Figure 7: HEAT MAPS FOR DERIVED LENET/MNIST FEATURE SPACES. *These deep features were taken from the* `ip1` *layer of the network after the ReLU activation. Rows represent feature vector elements while columns correspond to samples from the rotated MNIST test set.*

In Figure 6 we plot the scree diagram for each of our proposed approaches using `ip1` layer vectors extracted from the rotated MNIST test set as the data matrix for the SVD. From analyzing the scree diagrams, we see that the μ-LeNet representation is approximately rank-10, with little variance explained by subsequent components. This indicates not only that the network has converged well on the training set; it also indicates that the training set seems to have similar characteristics to the test set. Since the test set is basically balanced in terms of class labels, the sharp degradation in singular values suggests either 1.) that their respective singular vectors either do not represent the underlying digits but rather constituent parts of the underlying digits, or 2.) that some digits are more difficult to discriminate than others and require more variance to do so.

As a follow-up analysis we performed heat-map visualizations, depicting activations across the rotated test set. These heat maps are shown in Figure 7. In each of the figures, digit classes are sorted in ascending order (0 through 9) from left to right. Within each digit class, poses are sorted in ascending order ($-45°$ to $45°$). For all three of the feature spaces, we can discern ten modes corresponding to each of the digits, but the Raw-LeNet's feature space is far more scattered. Moreover, within each of the ten modes, we see a trend of amplification or attenuation moving from right to left. This suggests a noticeable dependence on pose in the Raw-LeNet representation. The μ-LeNet exhibits far more stability in the relative value of feature vector elements within a mode. The PAIR-LeNet representation fluctuates for a given element within a mode more than the μ-LeNet, resulting in noisier looking horizontal lines, but there is little visual evidence of clear-cut dependence on pose. Especially for the μ-LeNet, the fact that we see discernible horizontal line segments within each mode of the data matrix suggests that remaining pose-related information is present in low-magnitude noise.

Separate colorbars are shown for each plot in Figure 7 in order to get good relative visualization of the data matrices. Note that the Raw-LeNet's features are somewhat

greater in intensity than the PAIR-LeNet. The μ-LeNet's features are noticeably smaller in magnitude than either Raw-LeNet or μ-LeNet due to normalization required to get the distillation-like training to converge. Note that the PAIR-LeNet has a much sparser feature space than either of the other two architectures. We hypothesize that this is a result of the PAIR architecture's shared weights.

Another interesting question is: to what degree are individual feature vector elements associated with a particular class. To this end, we attempt to block-diagonalize each of the data matrices by performing row operations so as to maximize contrast. Specifically, given a $M \times N$ data matrix, where M is the feature dimension and N is the number of samples with known labels, we iterate through the M rows. For each unique label l in the set of possible labels $\mathbb{L}$ we assign error for the ith feature vector element and the lth label as:

$$E_{il} = \sum_{j=1}^{N} \frac{D_{ij}(1 - I(y_j, l))}{D_{ij} I(y_j, l)}, \tag{1}$$

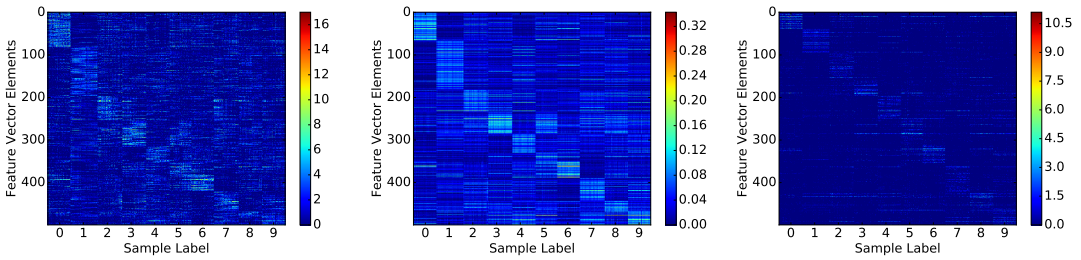
where $I(\cdot)$ is an indicator function that yields 1 if y_j is equal to l and 0 otherwise. Picking a cluster associated with the value of l for which (1) is minimized, and assigning the ith row of the matrix to that cluster for all rows ($i = 1, \ldots, M$), then vertically stacking the clusters from minimum to maximum l value yields the optimal block-diagonalization by the error measure in (1).

The respective block-diagonalizations are shown in Figure 8. While we can see a clear-cut diagonal structure in all three of the plots, the Raw-LeNet is far noisier than either the μ-LeNet or the PAIR-LeNet. The block diagonal structure of the PAIR-LeNet is faint due to sparsity, but has the fewest high-intensity off-block-diagonal elements. However, the few off-block-diagonal elements are highly saturated. While the μ-LeNet architecture has a strong block diagonal, we see that certain feature vector elements are easily confused between classes, e.g., those with high response for 3 and 5. Interestingly, far more features have high-response for 1 than any other class, even though classes are relatively well distributed. In all three cases, however, while there is clear-cut block diagonal structure, the strong co-occurrences of certain elements between classes suggest that enforcing saturation for a given class requires higher-level abstractions.

5 Conclusions and Future Work

The research and evaluations that we have conducted in this chapter suggest not only that one should expect the presence of non-invariances in deep feature spaces derived via common objective functions, but that attempting to attenuate such non-invariances and simultaneously maintain recognition performance is a challenging task, even with objectives explicitly designed to do so.

Generally, our analysis of the performance of face attribute prediction from face-identity derived deep features suggests what we would expect: that identity-related attributes tend



(a) Raw-LeNet (b) μ-LeNet (c) PAIR-LeNet

Figure 8: DIAGONALIZATIONS OF THE DATA MATRICES OF THE LENET/MNIST TEST SET. *Like Figure 7, these deep features were taken from the* `ip1` *layer of the network after the ReLU activation. Rows represent feature vector elements while columns correspond to samples from the rotated MNIST test set. In contrast to Figure 7 which displays rows in order, rows were re-ordered to best diagonalize the data matrix as per* (1).

to be more readily discriminated than identity-correlated attributes, which are in turn more easily discriminated than identity-independent attributes. However, the fact that second-stage classifiers were able to recognize all attributes with noticeably better than random performance dictates that there is still some information about those attributes contained in the deep feature representations, and consequently that the deep features are sensitive to variations in these attributes. Also, the fact that some identity-independent attributes (e.g., `Wearing Hat` or `Wearing Lipstick`) were very easily recognized suggests that – perhaps fundamentally – the representational capacity to best recognize an identity necessarily carries information that is highly relevant to recognizing the presence or absence of these non-identity-related attributes.

With respect to predicting the pose of a face, we find that across several network topologies trained for different tasks and on different datasets, pose information can be accurately recovered, even with simple linear regressors. We hypothesize that the reason the AFFACT and VGG network feature spaces offered most readily predictable pose information may have something to do with their lack of sensitivity to alignment, training across jittered data, and the generally high dimensionality of the feature space. Future research to ascertain which factors lead to pose discrimination could involve deepening these networks and reducing dimensionality by bottlenecking the output. In this chapter, we have not addressed how non-invariance changes as a result of network depth, which is a very relevant topic since adding layers to the network literally adds layers of abstraction.

The topology changes that we introduced to LeNet – both the μ-LeNet and the PAIR-LeNet exhibited some characteristics that we would expect from an invariant representation, but did not contribute noticeably to rotation-invariance, as our ability to recognize rotation barely diminished. Perhaps variations due to rotation (and pose) are very difficult

to attenuate. This would again be an interesting problem on which to explore the effects of deeper representations. Another interesting experiment would be to analyze the effects of using similar architectures to attenuate non-pose-related exogenous input variations.

Although the features extracted from the μ-LeNet still included information about pose, the scree diagram in Figure 6(b) suggests that this information is contained only in the part that is not varying much, i.e., it is not expressed strongly. Hence, as shown in Figure 4, this network achieved the best cosine-based classification accuracy. However, as the mean activation vectors are identically between training and test set, this result is surely biased. Translating this experiment to face recognition, the mean activation vector would be computed over training set identities, while evaluation would be performed on test set identities, which are different. In future work we will investigate if a μ-network approach would work in such a face recognition setting.

For the PAIR-LeNet, we only chose one pair of images in order to reduce the representational difference between those. While this reduced pair-wise distances, it did not work well to reduce class-wise distances. An interesting extension to the PAIR network would be to use several images of one class/identity at the same time and try to reduce batch-wise distances. With batches that are large enough and that are randomly selected across all training set images of one label in each iteration, we hope to reduce class-wise distances rather than pair-wise distances.

References

[1] Jun-Cheng Chen, Vishal M. Patel, and Rama Chellappa. Unconstrained face verification using deep CNN features. In *Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9. IEEE, 2016. 4

[2] Jun-Cheng Chen, Rajeev Ranjan, Amit Kumar, Ching-Hui Chen, Vishal M. Patel, and Rama Chellappa. An end-to-end system for unconstrained face verification with deep convolutional neural networks. In *International Conference on Computer Vision (ICCV) Workshop*, pages 360–368, 2015. 3, 4, 6, 7, 9, 10

[3] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. DeCAF: A deep convolutional activation feature for generic visual recognition. In *International Conference on Machine Learning (ICML)*, pages 647–655, 2014. 5

[4] Manuel Günther, Laurent El Shafey, and Sébastien Marcel. *Face Recognition Across the Imaging Spectrum*, chapter Face Recognition in Challenging Environments: An Experimental and Reproducible Research Survey. Springer, 1 edition, 2016. 9

[5] Manuel Günther, Andras Rozsa, and Terrance E. Boult. AFFACT - Alignment Free Facial Attribute Classification Technique. *arXiv preprint arXiv:1611.06158*, 2016. 3, 5, 7, 9, 10

[6] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 5, 14

[7] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007. 4

[8] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM International Conference on Multimedia (ACMMM)*. ACM, 2014. 11

[9] Brendan F. Klare, Ben Klein, Emma Taborsky, Austin Blanton, Jordan Cheney, Kristen Allen, Patrick Grother, Alan Mah, and Anil K. Jain. Pushing the frontiers of unconstrained face detection and recognition: IARPA Janus benchmark A. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2015. 4, 6

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 5

[11] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 4, 11

[12] Chi Li, Austin Reiter, and Gregory D. Hager. Beyond spatial pooling: fine-grained representation learning in multiple domains. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4913–4922. IEEE, 2015. 5

[13] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *International Conference on Computer Vision (ICCV)*, pages 3730–3738. IEEE, 2015. 3, 5, 7

[14] Connor J. Parde, Carlos Castillo, Matthew Q. Hill, Y. Ivette Colon, Swami Sankaranarayanan, Jun-Cheng Chen, and Alice J. O'Toole. Deep convolutional neural network features and the original image. *arXiv preprint arXiv:1611.01751*, 2016. 6

[15] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep Face Recognition. *British Machine Vision Conference (BMVC)*, 1(3):6, 2015. 3, 4, 6, 9, 10, 11

[16] Konda Reddy Mopuri and R. Venkatesh Babu. Object level deep feature pooling for compact image representation. In *Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 62–70, 2015. 3, 5

[17] Ethan M. Rudd, Manuel Günther, and Terrance E. Boult. MOON: A mixed objective optimization network for the recognition of facial attributes. *European Conference on Computer Vision (ECCV)*, 2016. 3, 5, 7

[18] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 2

[19] Swami Sankaranarayanan, Azadeh Alavi, Carlos D. Castillo, and Rama Chellappa. Triplet probabilistic embedding for face verification and clustering. In *Biometrics Theory, Applications and Systems (BTAS)*. IEEE, 2016. 4, 6

[20] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2014. 4, 6

[21] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *International Conference on Computer Vision (ICCV)*, pages 4068–4076, 2015. 5

[22] Jing Wang, Yu Cheng, and Rogerio Schmidt Feris. Walk and learn: Facial attribute representation learning from egocentric video and contextual data. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016. 5