Discriminatively Trained Autoencoders for Fast and Accurate Face Recognition

Paraskevi Nousi and Anastasios Tefas

Department of Informatics, Aristotle University of Thessaloniki
paranous@csd.auth.gr, tefas@aiia.csd.auth.gr

Abstract. Accurate face recognition is vital in person identification tasks and may serve as an auxiliary tool to opportunistic video shooting using Unmanned Aerial Vehicles (UAVs). However, face recognition methods often require complex Machine Learning algorithms to be effective, making them inefficient for direct utilization in UAVs and other machines with low computational resources. In this paper, we propose a method of training Autoencoders (AEs) where the low-dimensional representation is learned in a way such that the various classes are more easily discriminated. Results on the ORL and Yale datasets indicate that the proposed AEs are capable of producing low-dimensional representations with enough discriminative ability such that the face recognition accuracy achieved by simple, lightweight classifiers surpasses even that achieved by more complex models.

Keywords: Autoencoders, dimensionality reduction, face recognition.

1 Introduction

Face recognition is an important task in Machine Learning, and is vital in the problem of person identification as a person's facial image constitutes his/her most prominent biometric features [6]. Correctly identifying a person given a facial image may, for example, facilitate the task of automatic video capturing using Unmanned Aerial Vehicles. In such a scenario, after faces have been detected in the video stream provided by the UAV, the faces may be analyzed by a face recognition framework so as to identify persons of interest, e.g., important athletes in sports competitions. After correct identification has been established, the UAV may then use tracking algorithms to track a person, thus aiding the capture of opportunistic shots of this nature.

As face recognition is a very popular problem, many algorithms have been proposed for its solution over the past years. Autoencoders, in particular, have emerged as tools useful to this task and many variants have been proposed in literature, yielding impressive results. Typically, such methods on the one hand produce low-dimensional representations, lowering subsequent computational costs, but on the other hand, further analysis of the low-dimensional representations by computationally expensive classifiers is required, as lightweight

classifiers may not possess enough discriminative ability to produce accurate predictions. However, deploying such methods on UAVs, which are afflicted by limited computational capabilities, is inefficient. Intuitively, if the data representation obtained by an AE contains enough discriminative ability itself, even a simple, lightweight classifier should yield significant recognition accuracy.

This is the main intuition behind this work: we seek to incorporate label information into the training process of an AE, for the purpose of dimensionality reduction in conjunction with producing discriminative features so as to facilitate even lightweight classifiers. The representations produced by such an AE should be discriminative enough, such that a very simple classifier should be able to differentiate well between samples of different classes and a more complex classifier may be trained faster and more effectively. Furthermore, the compression of the data dimensionality reduces the computational costs imposed on the classifiers.

The main contribution of this work is the proposal of a method to implicitly incorporate label information into the training process of autoencoders, by shifting the data samples in the input space in a way such that they become more easily classifiable. The proposed method is lightweight, as it doesn't require any specialized loss functions or tuples of samples for the training process, making it suitable for deployment in mobile applications, e.g., on UAVs for identification of persons of interest. Moreover, significantly improved accuracy results are achieved in various classifiers, including computationally inexpensive ones such as the Nearest Centroid classifier.

The rest of this paper is structured as follows. Section 2 provides insight into related work on the subject of discriminative dimensionality reduction approaches and highlights the advantages of the proposed method. In Section 3, after a brief summary of autoencoders, the proposed discriminatively trained autoencoders are introduced. Section 4 presents the experimental setup used for the evaluation of the proposed method as well as the performance of various classifiers when using the proposed methodology. Finally, our conclusions are drawn in Section 5.

2 Related Work

Autoencoders [12] have been widely deployed in the past to facilitate several classification tasks, including the task of face recognition [10]. As autoencoders are trained in an unsupervised fashion, research interest steered towards incorporating supervised information into their training process, so as to produce hidden representations better suited to specific tasks.

In facial expression recognition, for example, [13] proposed a method to discriminate between features that are relevant to the facial expression recognition and features that are irrelevant to this purpose. In [8] the use of pose variations at given degrees of yaw rotation of the same face was suggested for mapping the variations back to the neutral pose progressively, by manipulating the loss functions for the variations.

In [14], the label information is incorporated into an AE's training process by augmenting the loss function so as to include the classification error. In another approach, [16] employ discriminative criteria by forcing pairs of representations corresponding to the same face to be closer together in the latent Euclidean subspace than to other representations corresponding to different faces, using a triplet loss method. However, heuristically selecting such triplets is very computationally expensive. Similarly, in [4], gated autoencoders, which require pairs of samples as inputs, were deployed for the task of measuring similarity between parents and children.

Methods such as the aforementioned ones focus on either incorporating the classification error into the AE's objective, or by utilizing carefully selected tuples of samples. In contrast, the method proposed in this work does not require any complex loss functions, which may disrupt the convergence of the reconstruction error of the AE, or the selection of any specialized tuples, which imposes heavy computational costs during the training process of the AE.

3 Proposed Method

In the following sections, a summary of autoencoders is presented. Then, by exploiting supervised information, the proposed discriminative autoencoders are introduced and analyzed.

3.1 Autoencoders

Autoencoders (AEs) are neural networks which learn to map their input into a latent subspace of typically lower dimension so as to reconstruct their input through the latent (or hidden) representation [17, 18]. As the input can be reconstructed given the hidden representation, the latter can be thought of as a low-dimensional representation of the input data.

The process through which the input is mapped to the latent representation is referred to as the encoding part of the AE and it may consist of several layers of neurons accompanied by non-linear activation functions. These non-linearities enable the AE to uncover more complex relations in the data, and separates autoencoders from linear dimensionality reduction algorithms.

Formally, an Autoencoder learns to map its input $\mathbf{x} \in \mathbb{R}^D$ into a hidden representation $\mathbf{y} \in \mathbb{R}^d$, using one or more layers of non-linearities:

$$\mathbf{y} = f(\mathbf{x}; \theta_{enc}) \tag{1}$$

where f denotes the encoding procedure and θ_{enc} is the set of parameters of the encoding part. The hidden representation $\mathbf{y}$ is then decoded through a similar procedure, i.e., one with a symmetrical architecture of layers, to produce the reconstruction $\hat{\mathbf{x}}$ of the input:

$$\hat{\mathbf{x}} = g(\mathbf{y}; \theta_{dec}) \tag{2}$$

where g denotes the decoding procedure and θ_{dec} is the set of parameters of the decoding part.

The parameters $\{\theta_{enc}, \theta_{dec}\}$ of the network are initialized either randomly or by using an improved initialization method [5], and updated through an error backpropagation algorithm, such as ADAM [9], so as to minimize the error between the produced reconstruction and the network's input, e.g., the mean squared error between the reconstruction and the input:

$$\ell = \|\hat{\mathbf{x}} - \mathbf{x}\|_2^2 \tag{3}$$

3.2 Discriminative Autoencoders

The latent representation produced by an AE is learned via minimizing the network's reconstruction error. Intuitively, if the target to be learned for each sample is a modified version of itself, such that it is closer to other samples of the same class, the network will learn to reconstruct samples which are more easily separable. This modification should intuitively be reflected by the intermediate representation, thus producing well-separated low-dimensional representations of the network's input data.

Let $\tilde{\mathbf{x}}_i^{(t)}$ be the target reconstruction of sample $\mathbf{x}_i$, then for $t = 0$, $\tilde{\mathbf{x}}_i^{(0)} \equiv \mathbf{x}_i$ corresponds to the standard AE targets. The *target shifting* process may be repeated multiple times, each time building on top of the previous iteration. The exponent t denotes the current iteration. The objective to be minimized for these *discriminative* autoencoders becomes:

$$\ell = \|\hat{\mathbf{x}} - \tilde{\mathbf{x}}^{(t)}\|_2^2 \tag{4}$$

The new targets may be shifted so that the samples are moved towards their class centroids, weighted by a small value α:

$$\tilde{\mathbf{x}}_i^{(t+1)} = (1-\alpha)\tilde{\mathbf{x}}_i^{(t)} + \alpha\Big(\frac{1}{|\mathcal{C}_i|}\sum_{\tilde{\mathbf{x}}_j^{(t)} \in \mathcal{C}_i} \tilde{\mathbf{x}}_j^{(t)}\Big) \tag{5}$$

where $\mathcal{C}_i$ is set of samples belonging to the same class as the i-th sample.

Respectively, the distances between samples and centroids of rival classes could be enlarged by moving each sample away from the mean of all samples belonging to other classes:

$$\tilde{\mathbf{x}}_i^{(t+1)} = (1+\alpha)\tilde{\mathbf{x}}_i^{(t)} - \alpha\Big(\frac{1}{N - |\mathcal{C}_i|}\sum_{\tilde{\mathbf{x}}_j^{(t)} \notin \mathcal{C}_i} \tilde{\mathbf{x}}_j^{(t)}\Big) \tag{6}$$

where N is the number of samples in the dataset. However, Equation (6) can be modified to only include the centroids of rival classes within a given range of each sample, or only the top k nearest rivalling centroids to the i-th sample, instead of the mean of all rival class samples.

4 Experiments

The proposed methodology is evaluated on two popular face recognition datasets, through measuring the accuracy achieved by several classifiers using as input the latent representations obtained by the discriminative AEs.

4.1 Datasets

The proposed methodology is evaluated on the ORL faces dataset [15] as well as the Extended Yale B dataset [11]. For both datasets, the grayscale images depicting the faces to be recognized are resized to 32×32, meaning the original data dimension is 1024. The dimension is downscaled by a factor of 4, down to 256, by the AEs.

The ORL dataset consists of 400 pictures depicting 40 subjects under slight pose, expression and other variations. Five-fold cross-validation is commonly used for the conduction of experiments with this dataset, where five experiments are conducted using 80% of the images per person as training data and selecting a different portion of the dataset for each fold such that all images serve as training and testing data at different runs.

The cropped version of the Yale dataset is used in our experiments, which contains images depicting 38 individuals under severe lighting variations and slight pose variations. Typically, half of the images per person are selected as training data and evaluation is performed on the remaining half images. We follow the same dataset splitting methodology performed five times and average the results over all folds.

4.2 Classifiers

The performance of four different classifiers is compared for the representations obtained by a classical AE and the representations obtained by the proposed AEs as well as for the original 1024-dimensional feature vectors corresponding to the pixel intensities.

Multilayer Perceptron A Multilayer Perceptron (MLP) [3], without hidden layers, maps its input to output neurons which correspond to the various classes describing the data. Thus the input layer has as many neurons as is the dimensionality of the input data, and the output layer has as many neurons as is the number of classes. The softmax function is typically used as the activation function in the output layer of neurons, in order to produce a probability distribution over the possible classes, which may then be used for the optimization of the network's parameters via the minimization of the categorical cross-entropy loss function.

Nearest Centroid The Nearest Centroid (NC) classifier assigns samples to the class whose centroid (i.e., mean of samples belonging to that class) lies the closest to them in space. The dimensionality of the data heavily affects the performance of this classifier, as the distances between very high-dimensional data have been shown to be inefficient for determining neighboring samples [1].

k-Nearest Neighbors Similarly to the NC classifier, the k-Nearest Neighbors (kNN) [2] classifier assigns samples to the class to which the majority of its k nearest neighbors belongs to. The dimensionality of the data affects the performance of this classifier as well, as it requires the computation of distances between all data samples.

Support Vector Machine A Support Vector Machine (SVM) [7] aims to find the optimal hyperplane to separate samples belonging to different classes. The kernel method can be utilized by SVMs to map the input data into a higher-dimensional space which is more easily separable by linear hyperplanes. In our experiments, the Radial Basis Function (RBF) kernel was used for this classifier.

4.3 Experimental Results

The accuracy achieved by the above classifiers is evaluated and compared for six types of inputs (parentheses show the respective notation used in corresponding Tables and Figures):

1. the original 1024-dimensional feature vectors, corresponding to pixel intensities (No AE)
2. the 256-dimensional latent representations achieved by a standard AE (AE)
3. the 256-dimensional latent representations achieved by the proposed AEs where the targets were shifted:
   (a) towards their class centers (dAE-1)
   (b) towards their class center as well as away from the nearest rival-class center (dAE-2)

For fair comparison between the results obtained by the standard AE and the proposed AEs the same architecture, number of epochs and initialization was used. In total, the target shifting process is applied five times over the training process of the AE. As for the hyperparameter α, a value of 0.4 was used for the shift towards the class centers and a small value of 0.01 for the shift away from rival centers, to ensure the shift in the input space is smooth.

The performance achieved by the evaluated classifiers for all input representations for the ORL dataset is summarized in Table 1. The dAE-2 method yields the best improvement for all classifiers, even though using the dAE-1 method the results still surpass those achieved by using the pixel intensities and the low-dimensional representations obtained by the standard AE.

Although the performance achieved by the MLP using the pixel intensities representation is quite high, the high-dimensionality of the data imposes higher

computational costs both in training and deployment. More importantly, the performance achieved by the less computationally intensive NC classifier is very close to the performance achieved by the MLP, and yields 10% and 15% improved accuracy results over the accuracy achieved when using the pixel intensities representation and the representation obtained by the standard AE respectively.

	MLP	NC	kNN	SVM
No AE	96.25 ± 1.58	85.25 ± 1.22	88.25 ± 5.51	90.75 ± 1.69
AE	92.75 ± 2.42	79.50 ± 1.87	82.25 ± 4.35	88.75 ± 2.50
dAE-1	96.00 ± 2.29	94.75 ± 2.29	95.25 ± 2.42	95.50 ± 1.69
dAE-2	$\mathbf{97.00 \pm 1.50}$	$\mathbf{95.50 \pm 1.87}$	$\mathbf{96.25 \pm 2.09}$	$\mathbf{96.50 \pm 1.87}$

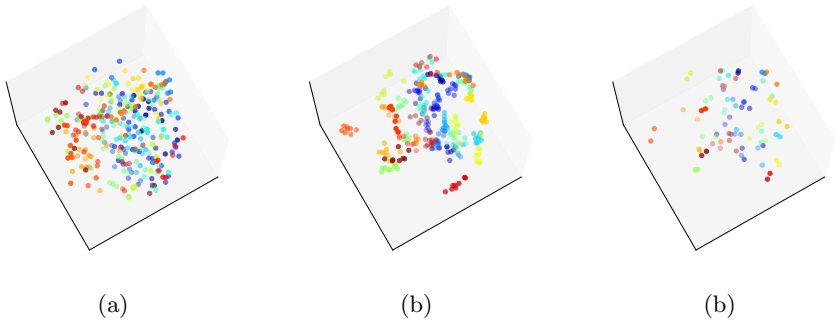
Table 1. ORL dataset accuracy results.

Table 2 summarizes the accuracy achieved by all classifiers and input representations for the Yale dataset. The proposed methods outperform the baselines by a large margin. The disadvantage of the NC and kNN classifiers when data dimensionality is high becomes very clear in this dataset when the pixel intensities are used as the data representation, indicated by their extremely inaccurate predictions and low accuracy results.

	MLP	NC	kNN	SVM
No AE	93.52 ± 1.07	10.94 ± 1.61	54.91 ± 1.54	71.19 ± 1.40
AE	88.25 ± 1.48	63.24 ± 2.06	73.70 ± 1.10	85.07 ± 1.48
dAE-1	$\mathbf{94.40 \pm 0.78}$	$\mathbf{89.93 \pm 0.68}$	91.12 ± 1.33	94.51 ± 0.63
dAE-1	94.30 ± 0.79	89.64 ± 0.91	$\mathbf{91.43 \pm 0.88}$	$\mathbf{94.61 \pm 0.74}$

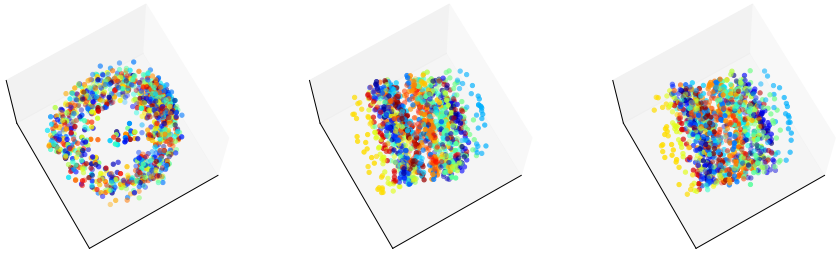
Table 2. YALE dataset accuracy results.

The results indicate that the proposed AEs are capable of generalizing well and implicitly applying the shifting process to unknown test samples. Figures 1 and 2 illustrate and ascertain this hypothesis. The left plot in both Figures is a 3-dimensional projection of the hidden representation learned by the standard AE, obtained via PCA. The middle plot is a 3-dimensional PCA projection of the representation learned by the dAE where the targets of the AE have been shifted five times in total towards their class centers. Finally, the plot on the right in both Figures is the 3-dimensional PCA projections of the hidden representation of the test samples, obtained by the same dAE as the middle projection. For both datasets, the 3D projections of the AE representation are difficult to unfold into separable manifolds. Through iterative repetitions of the target shifting process however, manifolds start to form and become obvious. Furthermore, the test samples appear at positions close to their counterparts used in the training process in the 3D projection, meaning that the AE learns to map those samples closer to their manifold.



(a) (b) (b)

Fig. 1. ORL hidden representation 3-dimensional projection by PCA: (a) hidden representation of the training data obtained by standard AE, (b) hidden representation of the training data obtained by dAE-1, and (c) hidden representation of the test data also obtained by dAE-1.



(a) (b) (c)

Fig. 2. YALE hidden representation 3-dimensional projection by PCA: (a) hidden representation of the training data obtained by standard AE, (b) hidden representation of the training data obtained by dAE-1, and (c) hidden representation of the test data also obtained by dAE-1.

The distribution shift that occurs to the training samples also affects the test samples belonging to the same classes. This is partly due to the fact that the testing samples follow more or less the distribution of the training samples in the input space as well. However, in the original input space as well as the subspace produced by the standard AE, the various class distributions are not well separated at all, which makes classification by lightweight classifiers very difficult. This is reflected by the extremely low accuracy results achieved by the NC and kNN classifiers especially in the Yale dataset (Table 2).

Figures 3 and 4 show samples (left) and their reconstructions (right) as given by the dAE-1 from the ORL and YALE dataset respectively. Figure 3a shows a sample from a training subset of the ORL dataset and its reconstruction. The pose variation of the input image is alleviated in the reconstruction, i.e. the face depicted is frontalized making it easier to recognize. Figure 3b shows a sample

from a test subset of the ORL dataset and its reconstruction, where the facial expression of the depicted face is neutralized.

(a) (b)

Fig. 3. Examples of ORL reconstructions: (a) input and reconstruction of a training sample, (b) input and reconstruction of a test sample.

Figure 4a shows a sample from a training subset of the YALE dataset and its reconstruction, as it is given by the dAE-1. As the sample is moved towards the centroid of its class, the illumination increases. Figure 4b shows a sample of a test subset of the YALE dataset and its reconstruction by the same AE. The network seems to have learned to generalize and is able to move the test sample towards its class centroid, removing the harsh obscurities imposed by the illumination imbalance.

(a) (b)

Fig. 4. Examples of YALE reconstructions: (a) input and reconstruction of a training sample, (b) input and reconstruction of a test sample.

The above reconstructions are consistent with the 3-dimensional projections shown in Figures 1 and 2 as well as the results presented in Tables 1 and 2 for the ORL and YALE datasets respectively: the discriminatively trained autoencoders are able to learn to map their input into a low-dimensional representation which is well-separated as well as to reconstruct a version of their input which is more informative about the depicted person's identity, by removing unrelated features such as pose, facial expression and illumination.

5 Conclusion

A method of training autoencoders, such that the low-dimensional representations of the data are more easily separable, has been proposed in this paper. The low-dimensional representation is learned in a way such that the reconstruction of the AE is a modified version of its input, which is shifted in space so that samples belonging to the same class will lie closer together and further from samples of rivalling classes. The proposed AE representations improve the performance of various classifiers, as illustrated by experimental results on two popular face recognition datasets. The classifiers' tolerance to pose, lighting and other variations is increased and they produce very accurate results while keeping the computational complexity low. Thus, the proposed AEs may be utilized in mobile environments, such as UAVs, for the task of fast and accurate person identification.

Acknowledgments

This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE). This publication reflects the authors views only. The European Commission is not responsible for any use that may be made of the information it contains.

References

1. Aggarwal, C.C., Hinneburg, A., Keim, D.A.: On the surprising behavior of distance metrics in high dimensional space. In: International Conference on Database Theory. pp. 420–434. Springer (2001)
2. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on pattern analysis and machine intelligence 19(7), 711–720 (1997)
3. Bhuiyan, A.A., Liu, C.H.: On face recognition using gabor filters. World academy of science, engineering and technology 28, 51–56 (2007)
4. Dehghan, A., Ortiz, E.G., Villegas, R., Shah, M.: Who do i look like? determining parent-offspring resemblance via gated autoencoders. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1757–1764 (2014)
5. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Aistats. vol. 9, pp. 249–256 (2010)
6. Goudelis, G., Tefas, A., Pitas, I.: Emerging biometric modalities: a survey. Journal on Multimodal User Interfaces 2(3), 217–235 (2008)
7. Guo, G., Li, S.Z., Chan, K.: Face recognition by support vector machines. In: Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on. pp. 196–201. IEEE (2000)
8. Kan, M., Shan, S., Chang, H., Chen, X.: Stacked progressive auto-encoders (spae) for face recognition across poses. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1883–1890 (2014)

9. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
10. Kotropoulos, C., Tefas, A., Pitas, I.: Frontal face authentication using variants of dynamic link matching based on mathematical morphology. In: Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on. vol. 1, pp. 122–126. IEEE (1998)
11. Lee, K.C., Ho, J., Kriegman, D.J.: Acquiring linear subspaces for face recognition under variable lighting. IEEE Transactions on pattern analysis and machine intelligence 27(5), 684–698 (2005)
12. Nousi, P., Tefas, A.: Deep learning algorithms for discriminant autoencoding. Neurocomputing (2017)
13. Rifai, S., Bengio, Y., Courville, A., Vincent, P., Mirza, M.: Disentangling factors of variation for facial expression recognition. Computer Vision–ECCV 2012 pp. 808–822 (2012)
14. Rolfe, J.T., LeCun, Y.: Discriminative recurrent sparse auto-encoders. arXiv preprint arXiv:1301.3775 (2013)
15. Samaria, F.S., Harter, A.C.: Parameterisation of a stochastic model for human face identification. In: Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on. pp. 138–142. IEEE (1994)
16. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 815–823 (2015)
17. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th international conference on Machine learning. pp. 1096–1103. ACM (2008)
18. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A.: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. Journal of Machine Learning Research 11(Dec), 3371–3408 (2010)