Multi-Task Adversarial Network for Disentangled Feature Learning

Yang Liu[1] Zhaowen Wang[2] Hailin Jin[2] Ian Wassell[1]
[1]University of Cambridge [2]Adobe Research
[1]{yl504,ijw24}@cam.ac.uk [2]{zhawang,hljin}@adobe.com

Abstract

We address the problem of image feature learning for the applications where multiple factors exist in the image generation process and only some factors are of our interest. We present a novel multi-task adversarial network based on an encoder-discriminator-generator architecture. The encoder extracts a disentangled feature representation for the factors of interest. The discriminators classify each of the factors as individual tasks. The encoder and the discriminators are trained cooperatively on factors of interest, but in an adversarial way on factors of distraction. The generator provides further regularization on the learned feature by reconstructing images with shared factors as the input image. We design a new optimization scheme to stabilize the adversarial optimization process when multiple distributions need to be aligned. The experiments on face recognition and font recognition tasks show that our method outperforms the state-of-the-art methods in terms of both recognizing the factors of interest and generalization to images with unseen variations.

1. Introduction

Image feature representation learning has been one of the central problems in Computer Vision. One of the most significant developments in the recent years in image feature learning is the resurgence of convolutional neural networks combined with large-scale datasets [14]. In this paper, we are interested in extending convolutional neural network based feature learning to the problems where multiple underlying factors determine the image generation process but only some factors are of our interest.

For many practical applications, the image generation process can be well approximated by a small number of factors. For instance, images of printed text are determined by factors such as font and glyph and images of human faces are determined by factors such as identity, pose, and illumination. We further assume there exists a primary factor for a given application. For instance, in the case of text images, if the application is font recognition, then font is the primary factor. But if the application is character recognition, then glyph is the primary factor. Multi-task learning [3] is the traditional approach to leverage the additional factors that are present in the image generation process. It learns a shared representation to predict all the factors. By doing so, we obtain features that can potentially outperform those learned from individual factors. However, if we are only interested in the performance of the primary factor, such as the identity for face images, can we do better than conventional multi-task learning?

Another major challenge in feature learning is generalization. We want the features learned from training data to perform well on test data that have never been seen in training. In the case of factored image generation processes, one particularly interesting generalization is to unseen variations of non-primary factors. For instance, if our problem is font recognition, we are interested in a feature representation that is robust to glyphs that have never been seen in training. Generalization is usually accomplished by seeing as many data variations as possible in training. However, in the case of images with factors, this would mean that we need to potentially see images with all the combinations of all the factors. We end up with an explosion of images (exponential with respect to the number of factors in the worse case). The interesting question is whether it is necessary to train on all the combinations of all the factors in order to generalize if there is a primary factor.

In this paper, we propose a novel feature learning algorithm for factored image generation processes that answers the previous two questions. Without loss of generality, we assume that the image generation process contains two uncorrelated factors and we are only interested in recognizing one of two. We refer to the factor of interest as the *content factor* and the other as the *style factor*. The key idea of the paper is that instead of learning from both factors in a cooperative way (traditional multi-task learning where both tasks help each other), we formulate the problem as learning from two *adversarial* tasks. To be more precise, given an input image with a content label and a style label, one task is to learn a content classifier and a shared image fea-

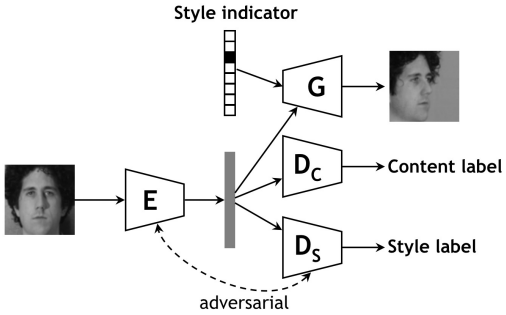

Figure 1. Overall network architecture of our proposed adversarial disentangling model. The model is composed of four components: an encoder, a generator, a content discriminator, and a style discriminator. The encoder is used to extract the content feature representation of the image, which is good for content recognition but not for differentiating the styles.

ture that label the image correctly according to the content label. The other task is to learn a style classifier and the same image feature that would label the image maximally incorrectly according to the style label. Through the adversarial process, we learn an image feature that outperforms that of multi-task learning on the content factor and generalizes to new images with both unseen content and style factors.

The overall framework of the proposed multi-task adversarial network (MTAN) is shown in Figure 1. There are four main components in our adversarial multi-task formulation: an encoder network, a generator network, and two discriminator networks. An input image is fed into the encoder network which produces the target feature representation. The feature is used as input to a content discriminator and a style discriminator. Both the encoder and the content discriminator work cooperatively to minimize a classification loss driven by the content label, while the encoder and the style discriminator play an adversarial game in which the interaction is modeled by a minimax optimization over the prediction of the style label. The two classification tasks are essentially competing with each other as the difference between the content and style classification losses is used to train the feature encoder. To ensure the encoded feature contains a full description of the image content, we also add a generator network to produce an image that matches the content of the input image and the style of a given style indicator. Depending on whether the style indicator matches the style label of the input image, the generator is trained to either reconstruct the input image or transfer it to a different style. By combining the encoder, the generator, and the two discriminators, we obtain a feature that is optimized with respect to the content factor while being style-agnostic. In this way the feature can generalize to unseen style factors without causing confusion on content understanding.

Same as other generative adversarial networks (GAN) [7, 1], the training process of the propose network architecture tends to suffer from unstable numerical optimization due to the minimax loss function. Moreover, in the problems we are interested in, the style discriminator may need to distinguish as many as hundreds or thousands of classes as opposed to a binary decision (real or fake) as in most existing GANs. If we break the multi-class problem into a set of binary classification problems, we are actually required to solve a set of minimax problems coupled by the same encoder, which is much more challenging than for GAN. To tackle the problem, we extend the Wasserstein GAN (WGAN) [1] algorithm to address the multi-class scenario, which significantly improves the training stability.

The main contributions of this paper are three-fold: 1. We propose a multi-task adversarial network that learns a disentangled feature representation through adversarial training of competing tasks on uncorrelated image factors. 2. We achieve stable optimization of multiple minimax losses by extending the WGAN algorithm [1] to the multi-class scenario. 3. Our feature representation outperforms standard cooperative multi-task learning methods, and achieves the state-of-the-art performance on the face recognition and font recognition datasets. Our approach can better generalize to unseen variations in both content factors and style factors.

2. Related Work

2.1. Disentangled Representation

There is a large quantity of literature concerning learning disentangled representations. The bi-linear model is among the first to separate the content and style in the underlying set of observations [24]. With the recent development of deep learning, auto-encoders [11, 10, 4] and Boltzmann machine [22] are adopted as regularizers to combine the discrimination and self-reconstruction criteria, thus discovering the factors of variation beside those relevant for classification. In particular, Predictability Minimization [23] and the Fair variational auto-encoder [16] encourage independence between different latent factors. In addition to reconstructing the input, [28, 19] synthesize other images with the same content but with a different style to implicitly disentangle features. With the help of GANs, the work of [17, 15, 6, 5] further explores the application of disentangled representations in computer graphics and video prediction.

Our proposed method differs from the previous methods by combining cross-style image generation with an adversarial training strategy to learn disentangled features. It is worth noting that although [25] also uses this combination for feature disentangling, our proposed multi-task adversarial network differs in the following respects. Since

our main goal is to improve content classification performance instead of synthesizing high-quality images, we employ multi-task adversarial training on the latent feature representation, instead of on the synthesized image and the real image. This is designed for explicitly learning such a disentangled latent feature that is good for content recognition but not for style recognition. By combining cross-style image generation, the learned feature is not only inclusive or generative for synthesizing a content-preserving image, but also exclusive or invariant to style variations, thus benefiting image classification.

2.2. Adversarial Learning

Adversarial training has been explored for representation learning in various computer vision applications. In most GANs, the aim is to minimize the divergence between the distribution of real and fake images. The similar adversarial training strategy has also been adopted in feature learning in domain adaptation [26] and video prediction [5]. However, these methods use binary adversarial objective functions, which means their adversarial training can only be generalized to the cases where data come from no more than two distributions. In contrast to their work, our proposed adversarial method considers multiple distributions. As shown in the adversarial branch in Figure 1, if we break the multi-class problem into a set of binary classification problems, we are actually required to optimize a set of minimax problems simultaneously, which are coupled by the same encoder.

It is worth noting that some works on GANs have claimed that they consider multiple categorical GANs, e.g., Semi-Supervised GAN [18] and DR-GAN [25]. Indeed, they have added a new branch for the multi-categorical classification, but in these previous approaches, the competing adversarial loss only confuses the discriminator by using two distributions (real or generated) and no adversarial strategies are adopted between different categories in the auxiliary multi-categorical classifier branch. In our work, the target of the encoder is to confuse the style classifier with any two classes, which aims to reduce the feature distribution discrepancy of any two style classes. There is no "real" reference class that can guide the distribution of other classes. In Sec. 3.2, we will provide an in-depth discussion concerning the difference between our proposals and most relevant work concerning conventional GANs.

It is also worth noting that min-max optimization always suffers from training instability as has been observed in related GAN research. In our model formulation, multiple min-max problems require to be optimized simultaneously, which further aggravates the training difficulties. Recent work in [1, 20] have been proposed, that are resilient to vanishing gradient and model collapse even with an overtrained loss function or mildly changed network architecture. In this way, the GAN network can be trained without properly balancing the generator and discriminator, which leads to improved training stability. However, these methods do not consider the adversarial loss for multiple distributions larger than two. Therefore, our work verifies whether these approaches can be extended to address multiple distributions.

3. Multi-Task Adversarial Network

The overall framework of multi-task adversarial network (MTAN) is illustrated in the Figure 1, where arrows indicate the forward propagation direction. It is composed of four components, including the encoder E, the generator G, the content classifier D_C and the style classifier D_S.

Assume we have an image $\boldsymbol{x}$ with discrete content label $y \in \mathcal{Y}$ and discrete style label $z \in \mathcal{Z}$. The image is first mapped by the encoder to its latent feature representation $E(\boldsymbol{x})$. Based on this latent representation, the discriminators try to predict class distributions $D_C(E(\boldsymbol{x})) \in \mathbb{R}^{|\mathcal{Y}|}$ and $D_S(E(\boldsymbol{x})) \in \mathbb{R}^{|\mathcal{Z}|}$ for content and style, respectively. As our goal is to encode image content information while removing any style variations in the learned feature representation, a good encoder E should extract a feature that is good for the content discriminator D_C but bad for the style discriminator D_S. Based on this intuition, we formulate the following adversarial multi-task training objectives:

$$\min_{E, D_C} \mathcal{L}_C \tag{1}$$

$$\max_{E} \min_{D_S} \mathcal{L}_S. \tag{2}$$

As can be seen, E and D_C work cooperatively to minimize the content classification loss $\mathcal{L}_C$, which is a conventional cross-entropy loss between ground truth y and prediction $D_C(E(\boldsymbol{x}))$. On the other hand, E and D_S play an adversarial game on the style loss $\mathcal{L}_S$, where E tries to minimize the divergence of feature distributions for different style classes so that D_S fails to correctly classify sample style no matter how hard it tries. Ideally, at the end of the competition, D_S can perform no better than a random guess. The idea behind (2) is the same as in GAN [7], although our goal is to learn disentangled features instead of generating images. Moreover, the style loss $\mathcal{L}_S$ typically involves a large number of classes as opposed to the binary classification in GAN. In our setting, there does not exist a "real" reference class that can guide the distribution of other classes; yet no trivial solution for E will be obtained as the encoder is also constrained by (1). The details of the loss function and training scheme for (2) will be given in Sec. 3.1.

Besides the multi-task classification branches, our network also includes a generation branch. The generator takes an encoded latent feature as well as a target style indicator z' as inputs, and outputs a synthesized image $G(E(\boldsymbol{x}), z')$

which shares the same content of $\boldsymbol{x}$ but is rendered in the style of z'. The training for the generation branch is guided by an $\mathcal{L}_2$ reconstruction loss:

$$\min_{E,G} \sum_{(i,j):y_j=y_i,z_j=z'_i} \|G(E(\boldsymbol{x}_i), z'_i) - \boldsymbol{x}_j\|_2 , \quad (3)$$

where subscripts i and j denote data indices. z'_i is randomly sampled from $\mathcal{Z}$. When $z'_i = z_i$, G tries to reconstruct the original input $\boldsymbol{x}_i$; otherwise, G generates a style-transferred version of $\boldsymbol{x}_i$ that matches a corresponding sample $\boldsymbol{x}_j$ with style z'_i in training database. The encoder-generator design contributes to content feature disentangling in an implicit way, and makes the encoded feature more inclusive of the image content.

3.1. Multi-Class Adversarial Training

Here we discuss the loss function and optimization strategy for the adversarial style classification in equation (2). The classification loss on a training pair $\{\boldsymbol{x}, z\}$ can be defined as the cross-entropy between predicted class distribution $D_S(E(\boldsymbol{x}))$ and ground truth label z:

$$\ell_{CE}(\boldsymbol{x}, z) = -\sum_{k\in\mathcal{Z}} \delta(z-k) \log D_S(E(\boldsymbol{x}), k), \quad (4)$$

where $\delta(\cdot)$ is the Dirac delta function, and $D_S(E(\boldsymbol{x}), k)$ denotes D_S's prediction score for the k-th style class. However, as pointed out in [1], cross-entropy is not a stable loss if there is a big disparity between the predicted distribution and target distribution. With the loss in (4), optimization in our case becomes even more unstable due to the large number of style classes and the absence of a fixed reference distribution.

Following the idea of WGAN [1], we improve optimization stability by replacing the cross-entropy loss with Earth Mover's Distance (EMD). As the goal of encoder E here is to match the feature distributions of multiple style classes, we need to calculate EMD for each pair of distinct styles. A more efficient alternative is to construct the pairs in a one-versus-all way, which gives the following multiple-distribution matching objective:

$$\min_E \sum_{k\in\mathcal{Z}} W\left(p(E(X)|Z=k), p(E(X)|Z\neq k)\right), \quad (5)$$

where $W(\cdot,\cdot)$ is the EMD distance, X and Z are the random variables for $\boldsymbol{x}$ and z. With the same approximation used in WGAN, the problem above can be converted to

$$\min_E \sum_{k\in\mathcal{Z}} \max_{D_S\in\mathcal{D}} \mathbb{E}_{\boldsymbol{x}\sim p(X|Z=k)} D_S(E(\boldsymbol{x}), k) - \mathbb{E}_{\boldsymbol{x}\sim p(X|Z\neq k)} D_S(E(\boldsymbol{x}), k), \quad (6)$$

where $\mathcal{D}$ is the space of all K-Lipschitz functions for some K. As D_S is shared by all the $|\mathcal{Z}|$ EMD operations, it is very hard to simultaneously achieve the optima for all the inner max problems. As an approximation, we switch the order of summation and maximization in (6), and arrive at our final objective function:

$$\min_E \max_{D_S\in\mathcal{D}} \sum_i -\ell_{EMD}(\boldsymbol{x}_i, z_i), \quad (7)$$

and

$$\begin{aligned}\ell_{EMD}(\boldsymbol{x}, z) &= -D_S(E(\boldsymbol{x}), z) + \frac{\sum_{k\neq z} D_S(E(\boldsymbol{x}), k)}{|\mathcal{Z}|-1} \\ &= <\boldsymbol{z}, D_S(E(\boldsymbol{x}))>,\end{aligned} \quad (8)$$

where $\boldsymbol{z} \in \mathbb{R}^{|\mathcal{Z}|}$ is a vector representation of z, with the z-th element equal to -1 and all the others equal to $1/(|\mathcal{Z}|-1)$. Note that when $|\mathcal{Z}| = 2$, our optimization objective reduces to the original WGAN.

To enforce the K-Lipschitz condition for D_S, we select $K = 1$ and adopt a gradient loss as in the improved WGAN [9] which is extended to multi-class as follows:

$$\mathcal{L}_R = \sum_{k\in\mathcal{Z}} \mathbb{E}_{\boldsymbol{u}}(\|\nabla_{\boldsymbol{u}} D_S(\boldsymbol{u}, k)\|_2 - 1)^2. \quad (9)$$

In practice, we sample latent feature $\boldsymbol{u}$ uniformly along the straight lines connecting pairs of training data $(E(\boldsymbol{x}_i), E(\boldsymbol{x}_j))$, where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are randomly sampled from training batch with different style labels: $z_i \neq z_j$. Note we interpolate feature points between any two style distributions rather than just two distributions as in the improved WGAN.

Finally, based on the loss terms in (8) and (9), we can formalize the style loss $\mathcal{L}_S$ as

$$\mathcal{L}_S = \sum_i \ell_{EMD}(\boldsymbol{x}_i, z_i) + \lambda\mathcal{L}_R, \quad (10)$$

where λ is a weighting parameter.

3.2. Comparison with Prior Adversarial Models

We compare MTAN with the three most relevant GAN and Auto-encoder variants as shown in Figure 2.

Semi-Supervised GAN: The Semi-Supervised GAN aims to learn a discriminative classifier where the discriminator D is trained to not only distinguish between real and fake, but also classify the real image in to K classes. D outputs a $(K+1)$-dim vector, in which the last dimension represents real/fake decision. The generator G aims to fool D by aligning two distributions, i.e., the distribution for real and fake images. MTAN differs to semi-supervised GAN in two aspects. Firstly, the input of the discriminator is the latent feature representation, instead of the real and synthesized image, and the goal of the encoder is to align the feature distributions between any two different style classes.

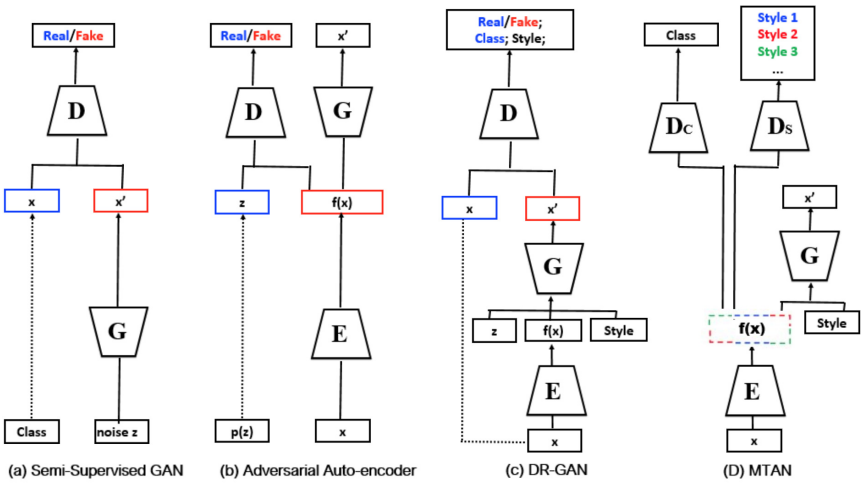

Figure 2. Comparison of previous GAN and auto-encoders architectures with our proposed MTAN.

Secondly, the encoder-generator structure learns a disentangled content representation implicitly by utilizing a target style indicator to generate images.

Adversarial Auto-encoder (AAE): In AAE, the auto-encoders $(E + G)$ reconstruct the input image, and the latent vector generated by the encoder matches an arbitrary prior distribution by training discriminator D. The MTAN model shares a similar image generation loss for generator but has two major differences. Firstly, besides the latent vector, we provide a target style indicator to the generator and generate a new image with the same content but in the style indicated. Secondly, we take advantage of an additional style variation classification task to disentangle the content-preserving feature representation explicitly.

DR-GAN: DR-GAN uses the encoder-generator structure to synthesize images. The goal of the encoder and generator is to fool Discriminator D to classify the synthesized image $\boldsymbol{x}'$ to the identity of input $\boldsymbol{x}$ and target style variation. Compared with MTAN, although they both adopt the encoder-generator structure for image synthesis, the goal of the discriminator and encoder is different in two respects. Firstly, the input of the discriminator is the extracted feature representation instead of the real and synthesized images. Secondly, the goal of the encoder is to extract such a latent feature representation, that contains little discriminative information about the style type, thus fooling the discriminator to make a random guess. In other words, the encoder needs to match or align the feature distribution between any two different style classes, instead of only real and fake distributions.

4. Experiments

In this section, we evaluate our feature disentangling method on two content-style image classification datasets, i.e., font and face datasets. We quantitatively evaluate the recognition accuracy using the disentangled representation as the content features with a cosine distance metric [21]. We also show qualitative results of synthetic images to demonstrate the learned feature is inclusive for generating content-preserving images.

4.1. Evaluation on Font Dataset

We compare our method with the various sub-models (that utilize selected elements of the MTAN model) to study the effectiveness and significance of each part of the MTAN model. We also analyze the training stability and generalization ability by testing on large or unseen variations.

4.1.1 Dataset and Experiment Setting

We have built a Japanese font recognition dataset. The reason we choose the Japanese language is that a large number of glyphs in the Japanese language introduce a large intra-class variation for the font recognition task, which makes the problem challenging. We have collected 300 fonts in total. We randomly split the data into 200 font classes for training and the remaining 100 font classes for testing. In the training stage, we use 50 frequently used glyphs as the style variation within each font class, and use the font file of a particular font to render this predefined glyph to form a training sample. We consider the following three settings to evaluate the generalization performance of our method when partial font classes or glyph styles are missing in the training stage. More specifically, one image per font category (with the same glyph) is randomly selected as gallery and the others are the probes throughout the paper unless otherwise specified.

Unseen Font: We test our model on the images from the remaining 100 unseen font classes, in which the glyph style is the same as that covered in the training stage.

Table 1. Recognition rate (%) comparison on Font database

Model	Unseen Font	Unseen Glyph	Unseen Both
CPF [28]	45.1	44.6	28.3
DR-GAN [25]	46.4	50.5	31.9
D_C	36.5	38.6	22.9
G	44.4	42.0	27.7
$D_{(C+S)}$	24.3	42.3	13.8
$D_{(C-S)}$	43.2	49.5	30.4
$D_{(C-S)}+G$	**47.9**	**52.8**	**34.8**

Unseen Glyph: We select another 50 glyphs, that are different from the training ones, and rendered them by the 200 seen font classes as the test set. We use the trained font classifier D_C for testing.

Unseen Font and Glyph: We select another 50 glyphs, that are different from the training ones, as the variation for the test set, and the images are rendered by other 100 unseen font classes.

4.1.2 Network structure and Implementation Details

For the encoder and generator, layer normalization is applied after each convolutional layer. Since the stability of the adversarial training suffers if sparse gradient layers are used, we replace MaxPool and ReLu with stride convolution and exponential linear unit respectively. Each discriminator (D_C and D_S) contains two fully connected layers. The output of the encoder is the disentangled content feature representation $f(x) \in \mathbb{R}^{512}$. $f(x)$ is then concatenated with a target glyph indicator. The generator contains a series of fractional-stride convolutions [2], which transforms the (512+$|\mathcal{Z}|$)-dim concatenated vector into a synthesized image, which has the same size as the original input image. The detailed information of the network structure is provided in the supplementary material.

We render all font images to size 96×96. The image intensities are linearly scaled to the range of [-1,1]. The batch size is set to be 256. An Adam optimizer [13] is used with a learning rate of 0.001 and momentum 0.5. We alternate between one step of optimizing the discriminators and generator, and one step of optimizing the encoder.

4.1.3 Results and Comparisons

We evaluate the font recognition performance under the three test settings and compare with the following prior work, i.e., Controlled Pose Feature (CPF) [28] and Disentangled Representation learning-Generative Adversarial Network (DR-GAN) [25]. We also present an ablation study for each module that we designed in our proposed method. Specifically, besides the models in [28][25], we also evaluate and compare with the following models:

1. **Single-Task** (D_C): trained on encoder and the font classifier $\boldsymbol{D_C}$ only using a soft-max cross entropy loss.
2. **Encoder-Generator** (G): trained on the encoder and the generator only using an $\mathcal{L}_2$ reconstruction loss for image generation.
3. **Multi-Task** ($D_{(C+S)}$): trained on encoder, font and glyph classifiers with conventional multi-task training to improve both the font and glyph recognition performance cooperatively.
4. **Adversarial-Task** ($D_{(C-S)}$): trained on encoder, font and glyph classifiers with adversarial training to a learn a disentangled representation as proposed in our model.
5. **MTAN** ($D_{(C-S)}+G$): trained on all modules designed in the proposed method, including the adversarial training and image reconstruction requirement to learn a disentangled representation.

The performance of the 7 chosen models are presented in Table 1. The single-task model trained on the encoder and font classifier only serves as the baseline. As shown in the table, introducing different combinations of the modules we designed all boost the performance under all three settings. It is worth noting that although the multi-task and the adversarial task model both leverage the font and glyph label supervision, the disentangled feature obtained from adversarial training performs much better than that achieved by conventional multi-task learning, especially when testing on unseen font. Combining the adversarial training and image reconstruction requirement together, MTAN achieves the best performance among these 7 models in all test settings. It is worth noting that although DR-GAN also takes advantages of this combination, our method outperforms DR-GAN by around 2%. This may be because the MTAN uses the multi-task adversarial training on the latent feature representation, instead of on the real and synthesized images, to disentangle the content feature explicitly. It makes the disentangled feature become more discriminant for the font classification task.

Another interesting observation is that, for the test on unseen font, when only adding one module into the single-task network, adding the generator module yields most of the available performance gain. It means that image generation makes the learned representation more inclusive or generative for synthesizing content-preserving images. The use of image generation also makes the learned feature applicable for extracting the content feature for the images coming from novel classes (not covered in the training stage). On the other hand, for the test on unseen glyph, when only adding one module into the single-task network, using the adversarial discriminators yields most of the available performance gain. It means that the multi-task adversarial training between the encoder and two discriminators acts as the critical role in learning a disentangled content representation that is exclusive or invariant to glyph variations. It is ideal for the font recognition, especially on images with large or unseen variations during training.

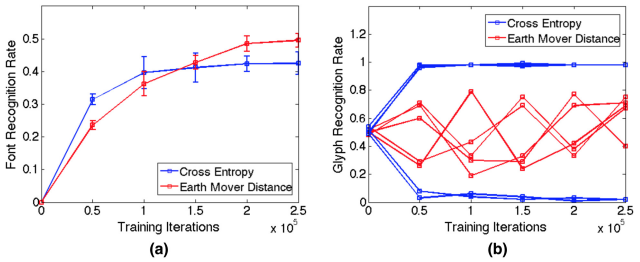

Figure 3. Comparison of font and glyph recognition accuracy on test set between models trained with different adversarial losses. (a) the mean and standard deviation (error bar) of font recognition; (b) the glyph recognition accuracy in multiple training trails.

4.1.4 Analysis

Training stability: We analyze the training stability for our proposed model and compare it with the model trained using cross-entropy as the style classification loss. Each model is trained for 5 times to get reliable observations. Figure 3 shows the accuracy for font recognition (a) and glyph recognition (b) on the test set achieved by the two models during training. Figure 3 (a) shows that using EMD loss consistently converges to a higher font recognition accuracy than using cross-entropy loss, although it converges slower. The error bars in (a) represent the standard deviations of test accuracy.

Figure 3 (b) plots the glyph recognition accuracy for all the training trials of the two models. The model trained with cross-entropy converges to either 1 or 0 quickly, indicating that the balance between E and D_S cannot be well maintained and the model collapses to local optimum. In contrast, the glyph accuracy of the model trained with EMD loss mildly oscillates around 50%, and the variation among different trials is small. In this way, the competition between E and D_S is more effective. Therefore, our proposed EMD loss can improve multi-class adversarial optimization stability, leading to better disentangled representation for font content.

Font image synthesis: Our generator is trained to synthesize new glyphs with the same font style as an input font feature. The feature could be calculated based on one input glyph image or the average of multiple glyphs from the same font. The identity of new glyph is specified by the style (glyph) indicator. Figure 4 shows some visualizations of the synthesized images. The synthetic glyphs are similar to the ground truth with well-preserved font attributes such as weights, and some fine-grained attributes like serif or sans-serif are also captured. The synthesized images demonstrate the learned disentangled feature includes most of the content-relevant information (font) which is faithfully reconstructed in the synthesized images.

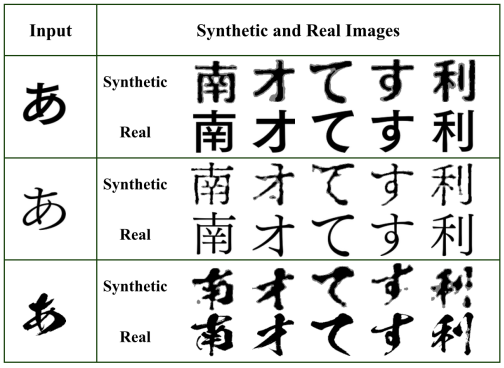

Figure 4. Synthetic images that matches the font of the input image and the style of given style indicators. We compare synthetic images (top) and their ground truth images (bottom).

4.2. Evaluation on Face Dataset

We evaluate our method on the face recognition dataset and compare with other state-of-the-art pose-invariant face recognition approaches. We also demonstrate our method can be used to disentangle more than one type of style variation.

4.2.1 Dataset and Experiment Setting

Multi-PIE [8] is a large database for evaluating face recognition under pose, illumination, and expression variations in a controlled setting. Following the setting in [29], we use 337 subjects with the neutral expression, 9 poses within 60^o, and 20 illuminations. The first 200 subjects are used for training and the remaining 137 for testing. For testing, one image per subject with the frontal view and neutral illumination is the gallery and the others are the probes. For Multi-PIE experiments, we disentangle more than two style factors, i.e., illumination and pose from the face identity. More specifically, we add additional pose and illumination codes as the input of the generator, and use two style discriminators to disentangle feature explicitly.

4.2.2 Network Structure and Implementation Details

To be consistent with the experimental setting of the comparison approaches, we adopt CASIA-NET [27] for the encoder and the generator design, where batch normalization and an exponential linear unit are utilized after each convolutional layer. The identity, pose and illumination discriminators are stacked after the encoder by adding a fully connected layer respectively. The output of the encoder is the identity representation $f(x) \in \mathbb{R}^{320}$, and this feature representation is then concatenated with a target pose indicator $\boldsymbol{z}_p \in \mathbb{R}^9$ and a target illumination indicator $\boldsymbol{z}_i \in \mathbb{R}^{20}$. Finally, the generator, which contains a series of fractional-stride convolutions [2], transforms the concatenated vec-

tor into a synthetic image. We follow the same data preprocessing as [27][25]. The batch size is 64, and all weights are initialized from the zero-centered normal distribution with a standard deviation 0.02. An Adam optimizer [13] is used with a learning rate of 0.0002 and momentum 0.5. We update G more frequently than D, i.e., 5 steps for optimizing the encoder and 1 step for the classifiers.

4.2.3 Results and Comparisons

In this section, we compare our proposed method with other existing pose-invariant face recognition approaches. The benchmark algorithms for comparison are Face Identity-Preserving (FIP) [29], multi-view perception (MVP) [30], multi-view deep network (MvDN) [12], Controlled Pose Feature (CPF) [28] and Disentangled Representation learning-Generative Adversarial Network (DR-GAN) [25].

Table 2 shows the face recognition performance on MultiPIE of our methods compared with the existing methods under the same setting, except for DR-GAN which uses multipe images for testing [25]. The components of the proposed MTAN are evaluated individually under the following settings:

1. **MTAN w/o D**: trained on the encoder and the generator only for image generation without using any adversarial training.

2. **MTAN1** ($D_{(C-S1)} + G$): trained on all modules, using adversarial training and having an image reconstruction requirement to disentangle two factors (identity and pose).

3. **MTAN2** ($D_{(C-S1-S2)} + G$): trained on all modules, using adversarial training and an image reconstruction requirement to disentangle three factors (identity, pose and illumination).

Our method shows a significant improvement for faces with extreme pose variations by disentangling the features using multi-task adversarial training and the image generation requirement. Compared with the other methods, the variation of recognition rates across different poses is much lower except DR-GAN, which suggests that our learned disentangled representation is more robust to the pose variation. The model MTAN2 which disentangles three latent factors further boosts the performance, which achieves comparable performance with the state-of-the-art performance, while not using multiple testing images as is done in DR-GAN. We also show some synthetic images in Figure 5. In the synthetic images, the identity of the input image can be faithfully preserved and the style is controlled by arbitrary style (pose, illumination) indicators. This means that the learned content (identity) representation is largely disentangled from other style variations (pose and illumination).

Table 2. Recognition rate (%) comparison on Multi-PIE database

Method	0^o	15^o	30^o	45^o	60^o	Avg.
FIP [29]	94.3	90.7	80.7	64.1	45.9	72.9
MVP [30]	95.7	92.8	83.7	72.9	60.1	79.3
MvDN [12]	96.1	93.1	83.3	75.1	61.2	80.1
CPF [28]	**99.5**	95.0	88.5	79.9	61.9	83.3
DR-GAN [25]	97.0	94.0	**90.1**	86.2	83.2	89.2
MTAN w/o D	94.1	92.7	83.7	72.9	60.1	79.3
MTAN1	95.2	93.2	88.9	84.7	82.6	88.9
MTAN2	96.5	**95.3**	89.7	**87.9**	**84.1**	**89.6**

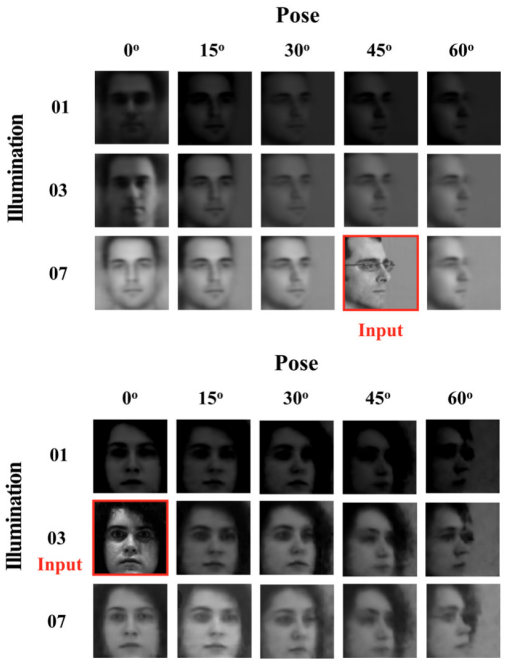

Figure 5. Face synthesis with varying poses and illuminations conditioned on single input image (indicated by red boxes).

5. Conclusion

In this paper, we propose a new deep network architecture based on a novel type of multi-task learning to disentangle image variation factors in the learned feature representation. The network includes an encoder-generator structure as well as a set of adversarial discriminators. Through the interaction with the discriminators and generator, the encoder learns to extract features good for content factor recognition but not useful for style factor recognition. The overall network can be trained stably with a new loss function which is an extension of WGAN for multi-class scenario. Quantitative and qualitative evaluation on both font and face datasets demonstrate the superiority of our proposed model over the state-of-the-arts.

References

[1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein GAN. *arXiv preprint arXiv:1701.07875*, 2017. 2, 3, 4

[2] D. Berthelot, T. Schumm, and L. Metz. BEGAN: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717*, 2017. 6, 7

[3] R. Caruana. Multitask learning. *Machine Learning*, 28, 1997. 1

[4] B. Cheung, J. A. Livezey, A. K. Bansal, and B. A. Olshausen. Discovering hidden factors of variation in deep networks. *arXiv preprint arXiv:1412.6583*, 2014. 2

[5] E. Denton and V. Birodkar. Unsupervised learning of disentangled representations from video. *arXiv preprint arXiv:1705.10915*, 2017. 2, 3

[6] A. Dosovitskiy, J. Tobias Springenberg, and T. Brox. Learning to generate chairs with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1538–1546, 2015. 2

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 2, 3

[8] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image and Vision Computing*, 28(5):807–813, 2010. 7

[9] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. Improved training of wasserstein GANs. *arXiv preprint arXiv:1704.00028*, 2017. 4

[10] G. E. Hinton, A. Krizhevsky, and S. D. Wang. Transforming auto-encoders. In *International Conference on Artificial Neural Networks*, pages 44–51. Springer, 2011. 2

[11] F. J. Huang, Y.-L. Boureau, Y. LeCun, et al. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. 2

[12] M. Kan, S. Shan, and X. Chen. Multi-view deep network for cross-view classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4847–4855, 2016. 8

[13] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6, 8

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 2012. 1

[15] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. Tenenbaum. Deep convolutional inverse graphics network. In *Advances in Neural Information Processing Systems*, pages 2539–2547, 2015. 2

[16] C. Louizos, K. Swersky, Y. Li, M. Welling, and R. Zemel. The variational fair autoencoder. *arXiv preprint arXiv:1511.00830*, 2015. 2

[17] M. F. Mathieu, J. J. Zhao, J. Zhao, A. Ramesh, P. Sprechmann, and Y. LeCun. Disentangling factors of variation in deep representation using adversarial training. In *Advances in Neural Information Processing Systems*, pages 5040–5048, 2016. 2

[18] A. Odena. Semi-supervised learning with generative adversarial networks. *arXiv preprint arXiv:1606.01583*, 2016. 3

[19] X. Peng, X. Yu, K. Sohn, D. N. Metaxas, and M. Chandraker. Reconstruction-based disentanglement for pose-invariant face recognition. *intervals*, 20:12, 2017. 2

[20] G.-J. Qi. Loss-sensitive generative adversarial networks on lipschitz densities. *arXiv preprint arXiv:1701.06264*, 2017. 3

[21] G. Qian, S. Sural, Y. Gu, and S. Pramanik. Similarity between euclidean and cosine angle distance for nearest neighbor queries. In *Proceedings of the 2004 ACM symposium on Applied computing*, pages 1232–1237. ACM, 2004. 5

[22] S. Reed, K. Sohn, Y. Zhang, and H. Lee. Learning to disentangle factors of variation with manifold interaction. In *International Conference on Machine Learning*, pages 1431–1439, 2014. 2

[23] J. Schmidhuber. Learning factorial codes by predictability minimization. *Neural Computation*, 4(6):863–879, 1992. 2

[24] J. B. Tenenbaum and W. T. Freeman. Separating style and content. In *Advances in neural information processing systems*, pages 662–668, 1997. 2

[25] L. Tran, X. Yin, and X. Liu. Disentangled representation learning GAN for pose-invariant face recognition. In *CVPR*, volume 4, page 7, 2017. 2, 3, 6, 8

[26] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. *arXiv preprint arXiv:1702.05464*, 2017. 3

[27] D. Yi, Z. Lei, S. Liao, and S. Z. Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014. 7, 8

[28] J. Yim, H. Jung, B. Yoo, C. Choi, D. Park, and J. Kim. Rotating your face using multi-task deep neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 676–684, 2015. 2, 6, 8

[29] Z. Zhu, P. Luo, X. Wang, and X. Tang. Deep learning identity-preserving face space. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 113–120, 2013. 7, 8

[30] Z. Zhu, P. Luo, X. Wang, and X. Tang. Multi-view perceptron: a deep model for learning face identity and view representations. In *Advances in Neural Information Processing Systems*, pages 217–225, 2014. 8