Label Denoising Adversarial Network (LDAN) for Inverse Lighting of Faces

Hao Zhou * Jin Sun* Yaser Yacoob David W. Jacobs
University of Maryland, College Park, MD, USA
{hzhou, jinsun, yaser, djacobs}@cs.umd.edu

Abstract

Lighting estimation from faces is an important task and has applications in many areas such as image editing, intrinsic image decomposition, and image forgery detection. We propose to train a deep Convolutional Neural Network (CNN) to regress lighting parameters from a single face image. Lacking massive ground truth lighting labels for face images in the wild, we use an existing method to estimate lighting parameters, which are treated as ground truth with noise. To alleviate the effect of such noise, we utilize the idea of Generative Adversarial Networks (GAN) and propose a Label Denoising Adversarial Network (LDAN). LDAN makes use of synthetic data with accurate ground truth to help train a deep CNN for lighting regression on real face images. Experiments show that our network outperforms existing methods in producing consistent lighting parameters of different faces under similar lighting conditions. To further evaluate the proposed method, we also apply it to regress object 2D key points where ground truth labels are available. Our experiments demonstrate its effectiveness on this application.

1. Introduction

Estimating lighting sources from an image is a fundamental problem in computer vision. In general, this is a particularly difficult task when the scene has unknown shape and reflectance properties. On the other hand, estimating the lighting of a human face, one of the most popular and well studied objects, is easier due to its approximately known geometry and near Lambertian reflectance. Lighting estimation can be used in applications such as image editing, 3D structure estimation, and image forgery detection. This paper focuses on estimating lighting from a single face image. We consider the most common face image type: near frontal pose. The same idea can be applied to face images with other poses.

There exist many approaches for lighting estimation from a single face image [6, 35, 19, 30], however they are not learning-based and rely on complicated optimization during testing, making the process inefficient. Moreover, the performance of these methods (e.g., [6]) depends on the resolution of face images, and cannot give accurate predictions for low resolution images.

Witnessing the dominant success of neural network models in other computer vision problems such as image classification, we are interested in a supervised learning approach that directly regresses lighting parameters from a single face image. Given an input face image, the approach outputs low dimensional Spherical Harmonics (SH) coefficients [8, 32]

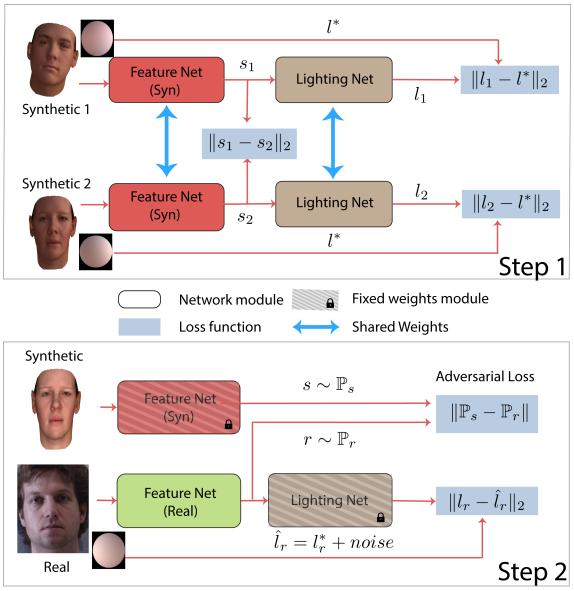

Figure 1. Training of a LDAN model has two steps: 1) Train the feature net and lighting net for synthetic data with two losses: Faces with similar lighting should have similar lighting related features ($||s_1 - s_2||_2$); Estimated lighting should be close to ground truth lighting ($||l_1 - l^*||_2$ and $||l_2 - l^*||_2$). 2) Train the feature net for real data while fixing both the feature net and lighting net trained in step 1. We use two losses in this step: The distribution of synthetic features and real features should be close ($||\mathbb{P}_s - \mathbb{P}_r||$); Estimated lighting should be close to noisy ground truth lighting ($||l_r - \hat{l}_r||$).

*means equal contribution.

of its environment lighting condition. This is a very difficult problem, especially due to the scarcity of accurate ground truth lighting labels for real face images in the wild. In fact, building a dataset with realistic images and ground truth lighting parameters is extremely hard and currently there exists no such dataset.

Lacking ground truth labels, we applied an existing method [6] to estimate lighting parameters of real face images. However, these lighting parameters are not the real "ground truth" as they contain unknown noise. Synthetic face images, on the other hand, have noise-free ground truth lighting labels. In this work, we show that this synthetic data with accurate labels can help train a deep CNN to regress lighting of real face images: "denoising" the unreliable labels.

The proposed method is based on two assumptions: (1) A deep CNN trained with synthetic data is accurate, i.e., it is not affected by any noise; (2) Ground truth labels for real data are noisy, but still contain useful information. We design the lighting regression deep CNN, which consists of two sub-networks: a feature net that extracts lighting related features and a lighting net that takes these features as input and predicts the Spherical Harmonics parameters. Based on the first assumption, the lighting net trained with synthetic data is accurate. However, this lighting net expects lighting related features for synthetic data as input. To make it work for real data, the lighting related features for real data should be mapped to the same space. For that purpose, we utilize the idea of Generative Adversarial Networks (GAN) [16]. Specifically, a discriminator is trained to distinguish between lighting related features from synthetic data and real data, while the feature net (instead of a generator in the standard GAN) is trained to fool the discriminator. The discriminator and our feature net play a minimax two player game, with the objective of pulling the distribution of lighting related features of real data towards that of the synthetic data. Under the second assumption, we have an additional objective of reducing regression loss between predicted lightings and ground truth labels. Moreover, we design the network to take 64×64 pixels RGB face images so that it will work for low resolution face images.

Figure 1 illustrates the proposed LDAN model. It consists of two steps during training: (1) Train with synthetic data; (2) Fix the feature net for synthetic data and the lighting net, train another feature net for real data with adversarial loss and regression loss. Eric et al. [40] proposed similar ideas, applying adversarial loss to map the distribution of features from the source domain to the target domain. However, they only use adversarial loss. We argue that such a mapping can be unpredictably arbitrary. As illustrated by Figure 2, both mapping A to A', B to B' and mapping A to B' and B to A' make the source and target data have similar distributions. This may be correct for classification tasks if

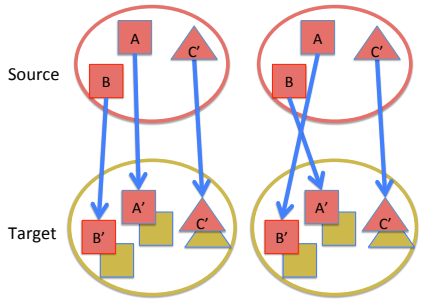

Figure 2. Two different functions that map data from the source domain to target domain with similar adversarial loss. With additional regression loss, our model is encouraged to learn a better behaved mapping function (the one on the left).

A and B belong to the same class. However, for regression, mapping A to B' makes the mapped feature far away from where it should be. As a result, using the regression loss for real data is critical in our regression problem: it regularizes the domain mapping function to have reasonable behavior. At the same time, the noise in real data labels are suppressed by training with the adversarial loss.

Since real ground truth labels for SH do not exist, we propose to use a classification based method to evaluate the consistency of estimated SH. However, this is still an indirect approach. To further evaluate the effectiveness of the proposed method, we apply it to an object key point regression problem where the ground truth labels are available. Similar to lighting regression from faces, we apply an existing method [42] to get the noisy ground truth and use synthetic data to help train an object key point regression network. Evaluated using the real ground truth, we demonstrate that LDAN works better than directly training a network with these noisy ground truth labels.

The main contributions of our work are: 1) We propose a lighting regression network for face images; 2) We propose a novel method, LDAN, to utilize accurate synthetic image lighting labels in training real face images with noisy labels; 3) The proposed method: increases the accuracy by 9% compared to [6] on quantitative evaluation, is robust to low resolution images, and is thousands of times faster.

2. Related Work

Lighting Estimation from A Single Face Image. Estimating lighting conditions from a single face image is a challenging problem. Blanz and Vetter [9] proposed to estimate the ambient and directional light as a byproduct of fitting 3D Morphable Models (3DMM) to a single face image. Since then, several 3DMM based methods were proposed [2, 30, 19, 35, 41, 12]. The performance of these methods rely on a good 3DMM of faces. However, existing 3DMMs are usually built with face images taken in a controlled en-

vironment, so their expressive power (especially the texture model) for faces in the wild is limited [10]. Barron and Malik proposed an optimization based method for estimating shape, albedo, and lighting for general objects [6]. To solve such an underconstrained problem, their method heavily relies on prior knowledge about shape, albedo, and lighting of general objects. Though they achieved promising results, their method is slow and may fail to give reasonable results in some cases due to the non-convexity of the objective function. [38] uses GRBMs to estimate albedo, normal, and illumination of a single image. They assume a distant point light source that is represented by the direction of the light which is less expressive than SH.

[24] proposed to use deep learning to disentangle representations about pose, lighting, and identity of a face image. The authors only show the effectiveness of their method on synthetic images; its performance on real face images is unclear. Recently, there is a trend to disentangle real faces using deep CNNs [36, 39, 23]. These methods, however, mainly focus on evaluating their performance on shape and albedo estimation. It is not clear whether the lighting estimated by these methods are accurate.

Learning with Noisy Labels. Learning with noisy labels has attracted the interest of researchers for a long time. [13] gives a comprehensive introduction to this problem. With the development of deep learning, many research studies have now focused on how to train deep neural networks with noisy labels [27, 37, 5, 43, 28, 21]. [27, 37, 28, 21] assume the probability of a noisy label only depends on the noise-free label but not on the input data, and try to model the conditional probability explicitly. [43] models the type of noise as a hidden variable and proposes a novel probabilistic model to infer the true labels. [5] proposes to use CNNs pre-trained with noise-free data to help select data with noisy labels in order to better handle the noise. All the above mentioned methods focus on classification problems and a considerable portion of the data are assumed to have noise-free labels. However, estimating lighting from face images is a regression problem, and the translation probability from noise-free labels to noisy labels is much more difficult to model. Moreover, the labels of our real data are noisy. As a result, we are dealing with a much harder problem than the methods mentioned above.

GAN for Domain Adaption. Since Goodfellow et al. [16] first proposed Generative Adversarial Networks, several works have been using this idea for unsupervised Domain Adaption [14, 40, 34, 33]. All these methods solve a problem in which the labels in the target domain are not enough to train a deep neural network. However, the problem we try to solve is intrinsically different from theirs in that the labels in the target domain are sufficient, but all these labels are noisy. Moreover, all these methods apply domain adaption to classification tasks where adversarial loss is enough to achieve a good performance. On the contrary, adversarial loss alone cannot work in our regression task. Though adversarial loss could map the distribution of data in the target domain to that of the source domain, for a single point in the target domain, the mapping is arbitrary which is problematic as every data point has its unique label in a regression task.

3. Proposed Method

3.1. Spherical Harmonics

[8, 32] have shown that for convex objects with Lambertian reflectance and distant light sources, the lighting of the environment can be well estimated by 9 (gray scale) or 27 (color) dimensions of Spherical Harmonics (SH). In this paper, we use SH as the lighting representation as it has been widely used to represent the environmental lighting in face related applications as suggested in [7, 41, 46, 6, 22, 30].

All dimensions of SH can be fully recovered from an image if the pixels are equally distributed over a sphere. However, the pixels of a face image, loosely speaking, are distributed over a hemisphere. The SH that can be recovered from a face image, as discussed in [31], lie in a lower dimensional subspace, and the SH for faces under different poses lie in different subspaces. As a result, we consider regressing the SH in a lower dimensional subspace instead of the original 27 dimensional SH and focus on near frontal faces since most face images are taken under this pose.

Taking the red color channel as an example, we now show how to get the lower dimensional subspace of SH for near frontal faces. Let $\mathbf{I}_r$ be a column vector: each element represents one pixel value of a face image for the red channel, then $\mathbf{I}_r = \Lambda_r Y \mathbf{l}_r$. Λ_r is a $n \times n$ diagonal matrix, each element of which is the albedo of the corresponding pixel, $\mathbf{l}_r$ is a 9 dimensional SH parameters vector, Y is a $n \times 9$ matrix and n is the number of pixels in the image. Each column of Y corresponds to one SH base image whose elements are determined by the normal of the corresponding pixel (see [8]). By applying SVD on Y, we get $Y = UDV^T$, then $\mathbf{I}_r = \Lambda_r U D V^T \mathbf{l}_r$. V is a 9×9 matrix that spans the entire 9 dimensions of SH. We use synthetic data to get V since we know the ground truth normal of every pixel and thus Y is known. We then only keep the first 6 columns of V, denoted as V_6, corresponding to the largest 6 singular values since they capture 99% energy of the singular values. With V_6, we project all the SH to their 18 dimensional subspace throughout the experiments.

3.2. Label Denoising Adversarial Network

Training a regression deep CNN needs a lot of data with ground truth labels. However, getting the ground truth lighting parameters from a realistic face image is extremely difficult. It usually needs a mirror ball or panorama camera

which is carefully set up to record an environment map relative to the position of the face. Instead, we adapted [6] to predict lighting parameters from a large number of face images. These parameters are then projected to a lower dimensional subspace using V_6 discussed above. We use these projected lighting parameters as noisy ground truth labels and denote them as $\hat{\mathbf{y}}_r$ for real face images $\mathbf{r}$. They are used as (data, label) pairs to train a deep regression CNN. Because these labels are noisy, directly training a deep CNN cannot give the best performance.

We propose to use synthetic face images, whose ground truth lighting parameters are known, to help train a better deep CNN model. The proposed model has two subnetworks: a feature network that is used to extract lighting related features and a lighting network that takes lighting related features as input and predicts SH. For synthetic data $\mathbf{s}$, we denote its feature network as $\mathcal{S}$ and its lighting network as $\mathcal{L}$. Then the predicted SH is represented as $\mathbf{y}_s = \mathcal{L}(\mathcal{S}(\mathbf{s}))$. Since $\mathcal{S}$ and $\mathcal{L}$ are trained using synthetic data with known ground truth labels, they are accurate. Feature network $\mathcal{R}$ and lighting network $\mathcal{L}_r$ for real data, on the other hand, are affected by noises if directly trained using the noisy ground truth of real data. To alleviate the effect of noisy labels, we propose to use $\mathcal{L}$ as the lighting net for real data, i.e., $\mathcal{L}_r := \mathcal{L}$, since it is not affected by noise. However, since $\mathcal{L}$ is trained using synthetic data, it only works if the input is from the space of lighting related features of synthetic data. As a result, $\mathcal{R}$ needs to be trained such that the lighting related features for real data are mapped into the same space as synthetic data.

Given a set of synthetic images $\mathbf{s}$ and their ground truth labels $\mathbf{y}_s^*$, we train feature net $\mathcal{S}$ and lighting net $\mathcal{L}$ through the following loss function:

$$\min_{\mathcal{S},\mathcal{L}} \sum_{(i,j)\in\Omega} \underbrace{[(\mathcal{L}(\mathcal{S}(\mathbf{s}_i)) - \mathbf{y}_{si}^*)^2 + (\mathcal{L}(\mathcal{S}(\mathbf{s}_j)) - \mathbf{y}_{si}^*)^2]}_{\text{regression loss for synthetic}} + \underbrace{\lambda(\mathcal{S}(\mathbf{s}_i) - \mathcal{S}(\mathbf{s}_j))^2}_{\text{feature loss}}, \quad (1)$$

where $\mathbf{s}_i$ and $\mathbf{s}_j$ are a pair of synthetic face images with the same SH lighting, different identities, and different small random deviations from frontal pose. $\mathbf{y}_{si}^*$ represents their ground truth label. Ω is a set containing all such pairs. λ is the weight for a feature loss. Besides the regression loss, we also add the MSE feature loss to enforce the lighting related features of face images with the same SH to be the same. This encourages the lighting related features to contain no information about face identities and poses.

With trained $\mathcal{S}$ and $\mathcal{L}$, next we train the feature net $\mathcal{R}$ for real face images $\mathbf{r}$ so that the lighting related features for real data ($\mathbf{f}_r = \mathcal{R}(\mathbf{r})$) lie in the same space as that of synthetic data ($\mathbf{f}_s = \mathcal{S}(\mathbf{s})$). Our idea is inspired by GAN [16]: a discriminator $\mathcal{D}$ is trained to distinguish $\mathbf{f}_r$ and $\mathbf{f}_s$, while $\mathcal{R}$ is trained so that $\mathbf{f}_r$ would make $\mathcal{D}$ fail. By playing this minimax game, the distribution of $\mathbf{f}_r$ will be close to that of $\mathbf{f}_s$. Wasserstein GAN (WGAN) [3] is used as our training strategy since it can alleviate the "mode dropping" problem and generate more realistic samples for image synthesis. However, making the distribution of $\mathbf{f}_r$ and that of $\mathbf{f}_s$ similar is not enough for our regression problem since the mapping can be unpredictably arbitrary. As shown in Figure 2, both mappings would make two sets of points have similar distributions, but they are not equally correct if we care about accuracy on individual points' labels. To deal with this problem, we use the noisy ground truth of real data as "anchor points" during training. As a result, the loss function for training on real data is defined as follows:

$$\min_{\mathcal{R}} \max_{\mathcal{D}} \underbrace{\sum_i (\mathcal{L}(\mathcal{R}(\mathbf{r}_i)) - \hat{\mathbf{y}}_{ri})^2}_{\text{regression loss for real}} + \mu \underbrace{(\mathbb{E}_{\mathcal{S}(\mathbf{s})\sim\mathbb{P}_s}[\mathcal{D}(\mathcal{S}(\mathbf{s})] - \mathbb{E}_{\mathcal{R}(\mathbf{r})\sim\mathbb{P}_r}[\mathcal{D}(\mathcal{R}(\mathbf{r}))])}_{\text{adversarial loss}} \quad (2)$$

where $\mathbb{P}_s$ and $\mathbb{P}_r$ are the distributions of lighting related features for synthetic and real images respectively.

Following [16, 3], the discriminator $\mathcal{D}$ and feature net $\mathcal{R}$ are trained alternatively. While training $\mathcal{D}$, RMSProp [20] is applied and Adadelta [45] is used to train $\mathcal{S}$, $\mathcal{R}$ and $\mathcal{L}$ as discussed in [3]. The details on how to train the whole model are illustrated in Algorithm 1.

Algorithm 1 Training procedure for LDAN

1: Train $\mathcal{S}$ and $\mathcal{L}$ for synthetic data using loss function in Equation 1 by Adadelta.
2: Compute lighting related features for synthetic images using $\mathbf{f}_{si} = \mathcal{S}(\mathbf{s}_i)$.
3: **for** number of training epochs **do**
4: **for** k=1 to 1 iterations **do**
5: Sample 128 $\mathbf{f}_s$ and $\mathbf{r}$. Train discriminator $\mathcal{D}$ through the following loss using RMSProp:

$$\max_{\mathcal{D}} \mathbb{E}_{\mathbf{f}_s\sim\mathbb{P}_s}[\mathcal{D}(\mathbf{f}_s)] - \mathbb{E}_{\mathcal{R}(\mathbf{r})\sim\mathbb{P}_r}[\mathcal{D}(\mathcal{R}(\mathbf{r}))]$$

6: **end for**
7: **for** k=1 to 4 iterations **do**
8: Sample 128 $\mathbf{r}$ and train $\mathcal{R}$ through the following loss using Adadelta:

$$\min_{\mathcal{R}} \sum_i (\mathcal{L}(\mathcal{R}(\mathbf{r}_i)) - \hat{\mathbf{y}}_{ri})^2 - \mu\mathbb{E}_{\mathcal{R}(\mathbf{r})\sim\mathbb{P}_r}[\mathcal{D}(\mathcal{R}(\mathbf{r}))]$$

9: **end for**
10: **end for**

Table 1. Accuracy of different methods. Standard deviation is shown in the bracket for learning based methods.

	SIRFS log	SIRFS SH	3DMM	REAL	LDAN	Model B	Model C
top-1 (%)	60.72	56.04	49.08	61.29 ($\pm$1.8)	**65.73** ($\pm$1.78)	56.62 ($\pm$3.86)	63.03 ($\pm$0.91)
top-2 (%)	79.65	74.39	65.78	81.95 ($\pm$1.3)	**84.57** ($\pm$1.35)	76.94 ($\pm$4.10)	82.79 ($\pm$0.35)
top-3 (%)	87.27	83.74	74.37	90.59 ($\pm$0.7)	**92.43** ($\pm$0.59)	86.69 ($\pm$3.39)	91.21 ($\pm$0.47)

4. Experiments

4.1. Data Collection

Real Face Images: The proposed LDAN requires a large number of both synthetic and real face images for training. For real face images, we download them from the Internet. SIRFS [6] is then applied to these face images to get the noisy ground truth SH. Since SIRFS was proposed to estimate lighting for general objects, their prior is not face-specific. To get a better constraint for a face shape, we apply Discriminative Response Map Fitting [4] to estimate the facial landmarks and pose. Then, a 3DMM [9] is fitted to estimate the face depth map which is used as a prior to constrain the face shape estimation of SIRFS. We collected $40,000$ faces with noisy ground truth SH for training.

Synthetic Face Images: We apply a 3D face model [29] to generate $40,000$ pairs of faces. Each pair of these faces are under the same lighting but with different identities and a small random variation with respect to frontal pose.

MultiPie: The MultiPie dataset [17] contains a large number of face images of different identities taken under various poses and illumination conditions. From this data set, $4,980$ face images are chosen, which contain 250 identities in frontal pose under 19 lighting conditions. Though the ground truth lighting parameters are not provided for each of these face images, the lighting condition group under which a face image is taken is given. This data is used only for evaluation in our experiments.

4.2. Implementation Details

We apply the same ResNet structure [18] for feature net $\mathcal{S}$ and $\mathcal{R}$. It takes a 64×64 RGB face image as input and outputs a 128-D feature vector. We define the lighting net $\mathcal{L}$ and discriminator $\mathcal{D}$ to be 2 and 3 fully connected layers respectively. The lighting net outputs 18 dimensional lighting parameters and $\mathcal{D}$ outputs the score for being a lighting related feature of real data. Please refer to the supplementary material for details on the network structures.

While training the proposed model, we first train discriminator $\mathcal{D}$ for 1 iteration and then train feature net $\mathcal{R}$ for 4 iterations. We alternate these two steps for 10 epochs. We choose $\mu = 0.01$, and $\lambda = 0.01$. Our algorithm is implemented using Keras [11] with Tensorflow [1] as backend.

4.3. Evaluation Metric

Since ground truth lighting parameters for real face images are not available, it is difficult to evaluate the accuracy of regressed lighting quantitatively. We propose an *indirect* classification-based metric and test our method on the MultiPie data set, which contains face images taken under 19 lighting conditions. More specifically, after regressing the SH for each test face image, 90% of them are used to compute the mean SH for each lighting condition group. Then, the rest images are assigned to the 19 lighting conditions based on the Euclidean distance between its estimated SH and the mean SH. We carry out 10 cross validations for this classification measurement to make use of all the data.

4.4. Experimental Results

We compare LDAN with SIRFS [6] based method in this section. In SIRFS, the shading of a face is formulated in logarithm space, i.e. $\log\{s_i\} = Y_i\mathbf{l}$ where s_i is the shading at the i-th pixel, Y_i is the i-th row of Y and $\mathbf{l}$ represents the SH in logarithm space. To estimate the correct SH lighting, we assume that the normal of each pixel estimated by SIRFS is in Euclidean space. Supposing $\hat{\mathbf{l}}$ is the correct SH, the shading can be found by $s_i = Y_i\hat{\mathbf{l}}$. Then $\hat{\mathbf{l}}$ can be found by solving the following overcomplete linear equation:

$$Y\hat{\mathbf{l}} = \exp\{Y\mathbf{l}\}. \tag{3}$$

Compare with baselines. Table 1 compares the proposed method with some baseline methods using the classification metric on MultiPie. We denote the original output of SIRFS as SIRFS log, and the corrected SH by Equation (3) as SIRFS SH. We test these two methods on the original resolution of the MultiPie data which is roughly 220×270 after cropping the faces. For comparison, we also show the results of a 3DMM model based lighting estimation method [15] (denoted as 3DMM). We notice that 3DMM performs worse than SIRF SH; this inspires us to use SIRF SH as the noisy ground truth for the real data. REAL in Table 1 represents a baseline method which uses SIRFS SH as ground truth to train a deep CNN without synthetic data. REAL and LDAN are trained 5 times and the mean accuracies are shown in Table 1. We notice that SIRFS SH performs worse than SIRFS log. This is because the accuracy of SIRFS SH depends not only on the accuracy of SIRFS log, but also on the accuracy of estimated normals. The noisy estimation of normals makes the SIRFS SH less reliable. The performance of REAL is better than SIRFS SH, though it is trained directly using the output of SIRFS SH as the ground truth label. This suggests that by observing a large amount of data, the deep CNN itself can

Table 2. Results of ablation study. Standard derivation is shown in the bracket.

	LDAN	LDAN w/o Adversarial	LDAN w/o Regression	LDAN w/o Fixed Lighting Net
top-1 (%)	**65.73** ($\pm$1.78)	63.63 ($\pm$2.12)	30.72 ($\pm$0.63)	63.95 ($\pm$0.60)
top-2 (%)	**84.75** ($\pm$1.35)	83.44 ($\pm$1.57)	49.12 ($\pm$0.85)	83.97 ($\pm$0.25)
top-3 (%)	**92.43** ($\pm$0.59)	91.48 ($\pm$1.09)	61.58 ($\pm$1.07)	92.07 ($\pm$0.46)

be robust to noise to some extent. This shows an advantage for learning based methods compared with optimization based algorithms. LDAN outperforms REAL by more than 4% and SIRFS SH by more than 9% for top-1 accuracy, showing the effectiveness of the proposed pipeline.

We further propose two other baselines to compare with LDAN as shown in Figure 3. Different from LDAN, Model B and Model C learn the feature nets for synthetic and real data simultaneously and map the lighting related features of them to the same space. These two models are inspired by [14] and [33]. For Model B, synthetic and real data share the same feature net. Since synthetic data and real data are quite different from each other, using a single feature net, it is difficult to make their lighting features have the same distribution, and we do not expect good performance. Model C defines different feature nets for synthetic and real data. The difference between Model C and LDAN is that Model C tries to map lighting related features for synthetic and real data to a common space, which might be different from that learned with synthetic data alone, whereas LDAN tries to directly map lighting related features of real data to the space of synthetic data. Intuitively, compared with LDAN, Model C is more easily affected by the noisy labels of real data since the training of the feature net for synthetic data is affected by the real data.

Model B and C are also trained 5 times and their mean accuracies are shown in Table 1 for comparison. We notice that Model B performs even worse than REAL, which shows that using the same feature net for both synthetic and real data is not a good idea. LDAN and Model C outperform all other methods in Table 1. Moreover, LDAN performs better than Model C, showing that it is more robust to the noise in the real data labels.

Ablation Study. To investigate the effectiveness of adversarial loss and regression loss, we carry out ablation studies for LDAN. We train the feature net 5 times for real data without adversarial loss or regression loss respectively and compare the results with LDAN in Table 2. Without adversarial loss, the performance of LDAN is better than REAL in Table 1, which means that synthetic data can help to regress lighting in this case. Without regression loss, on the other hand, the performance of LDAN drops dramatically. This is because the mapping of the distribution of lighting related features of real data to that of synthetic data is arbitrary as shown in Figure 2. This is problematic for a regression task where each data has its unique label. Having noisy ground truth as "anchor points", as we do in LDAN, can alleviate this problem and give much better results. We also train LDAN without fixing the lighting net and show the results in Table 2. We notice that the performance is similar to training LDAN without adversarial loss. This is expected since the lighting net are trained to adapt to the noisy labels: the impact of the adversarial loss is reduced.

Table 3. Accuracy of LDAN for different scale of face images.

Method	LDAN			SIRFS SH
Resolution	64×64	32×32	16×16	64×64
top-1 (%)	**65.73**	64.89	61.72	42.17
top-2 (%)	**84.75**	84.39	82.17	61.94
top-3 (%)	**92.43**	92.10	90.94	74.51

Visualizing Estimated Lighting. Figure 4 and Figure 5 visualize the SH parameters estimated by SIRFS, and LDAN from MultiPie images and the CelebA [26] data set respectively. Though there are few images with strong side light effect, we notice that LDAN can still work reasonably well for such images as shown in Figure 4 (b) and (d). However, the predicted lightings are not as sharp as those by SIRFS. This is mainly because the performance of learning based methods are heavily dependent on the training data. Without sufficient face images with strong side light for training, the performance of LDAN on those images may not be optimal. In classification we expect extreme lighting to be differentiated more easily than normal lighting because equal changes in the angle of lighting directions affect frontal lighting less than side lighting under the Lambertian model (details discussed in supplementary material). We notice that the lighting predicted by SIRFS can have incorrect directions (Figure 5 (a) (b) (c) and (d) as well as Figure 4 (c)). One of the reasons is the effect of the hair. Since the facial landmark detection method is not perfect, some of the hair regions are included in the cropped face images, which confuses SIRFS. Moreover, some lighting predicted by SIRFS have the incorrect color tone, especially for faces with dark reflectance, as shown in Figure 5 (e) (f) (g) and (h). On the other hand, LDAN is not affected by these two issues.

Robustness to Low Resolution Images. To investigate the robustness of the proposed method for low resolution images, we downsample face images of MultiPie to 32×32 and 16×16 and then resize them to 64×64 and evaluate the lighting classification accuracy using our trained LDAN model. As shown in Table 3, our trained model is quite robust to low resolution images, even for face images with

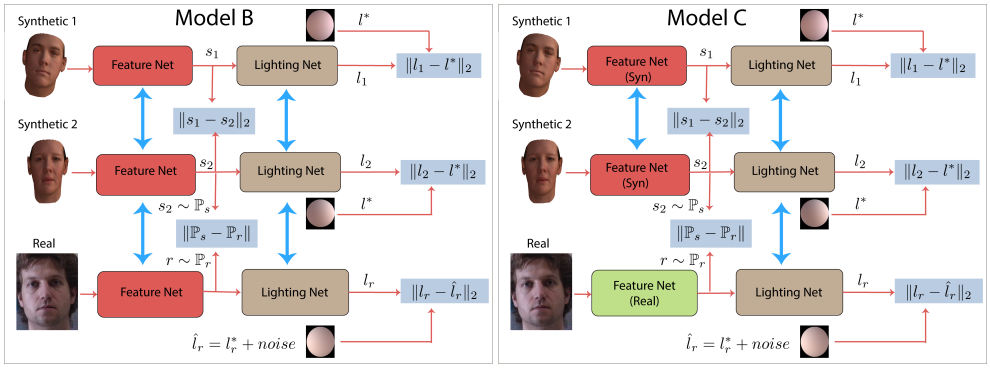

Figure 3. Two models we use to compare with the proposed LDAN. Different from LDAN, Model B use the same feature net for synthetic and real data; Model C trains feature net for synthetic and real data together.

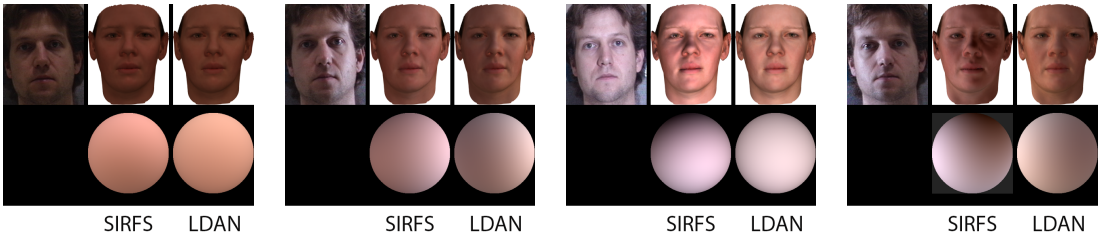

Figure 4. First row: MultiPie face image, rendered synthetic face with SIRFS estimated lighting and rendered synthetic face with LDAN estimated lighting. Second row: the hemisphere visualization of the corresponding estimated lightings. Images are best viewed on screen.

size 16×16, the top-1 accuracy only drops 4% compared with the original resolution (64×64). To compare, we also run SIRFS on 64×64 face images. Since we cannot run 3DMM on lower resolution images to get a good initialization, we fit the 3D model on the original resolution and resize it accordingly. We notice that the performance of SIRFS drops drastically (14%) even on 64×64 images.

Denoising Effect of LDAN To check the denoising effect of LDAN, we carried out experiments with synthetic data. We select $35,000$ synthetic images as the noise-free training data, $15,000$ as the noisy training data, and $5,000$ as testing data. We add three levels of Gaussian noise (std 0.1, 0.2, and 0.5) to the ground truth SH of synthetic data to prepare the noisy training data. To evaluate the performance, we render pairs of face images using the LDAN estimated lighting and the corresponding groundtruth lighting and compute their per-pixel error for evaluation. Table 4 shows results compared with a directly trained network. As a reference, the per-pixel error of directly training the network using noise-free data is 0.0714. The results suggest that LDAN can indeed reduce the impact of noisy labels.

Table 4. Per-pixel error for synthetic data with different noise.

	directly train	LDAN
Std 0.1	0.0893	**0.0740**
Std 0.2	0.1158	**0.0738**
Std 0.5	0.2531	**0.1752**

Running Time. We run experiments on a workstation with 4 Intel Xeon CPUs and 80 GB memory. While running on a GPU, we use one NVIDIA GeForce TITAN X. For a 64×64 RGB face image, SIRFS [6] takes 47 seconds. The proposed deep CNN can predict 390 such face images on the CPU and $2,400$ face images on the GPU per second, so it is potentially $100,000$ times faster than an optimization based method such as SIRFS.

4.5. Object 2D Keypoints Detection

Since ground truth lighting is hard to obtain, to better quantitatively check the effectiveness of the proposed method, we apply the LDAN training strategy to the object 2D keypoint detection problem, which has ground truth labels. The keypoint-5 dataset provided by 3DINN [42] has ground truth labels for 2D keypoints of sofa, chair, bed and swivel chair. [25] provided synthetic images of sofa and chair and their corresponding labels. Since 3DINN has achieved very high accuracy on the chair data set, we focus on using sofa to test our method.

To mimic noisy labels, we apply the code provided by 3DINN to predict the keypoints of sofa, and then double the noise of these labels so they contain more noises. Supposing $\mathbf{l}^*$ is a ground truth location of a keypoint (2 dimensional vector) and $\mathbf{l}'$ is the keypoint location predicted by 3DINN, we double the noise in the label $\mathbf{l}'$ and get $\hat{\mathbf{l}} = 2\mathbf{l}' - \mathbf{l}^*$. Unless otherwise specified, **we use $\hat{\mathbf{l}}$ as noisy labels to train**

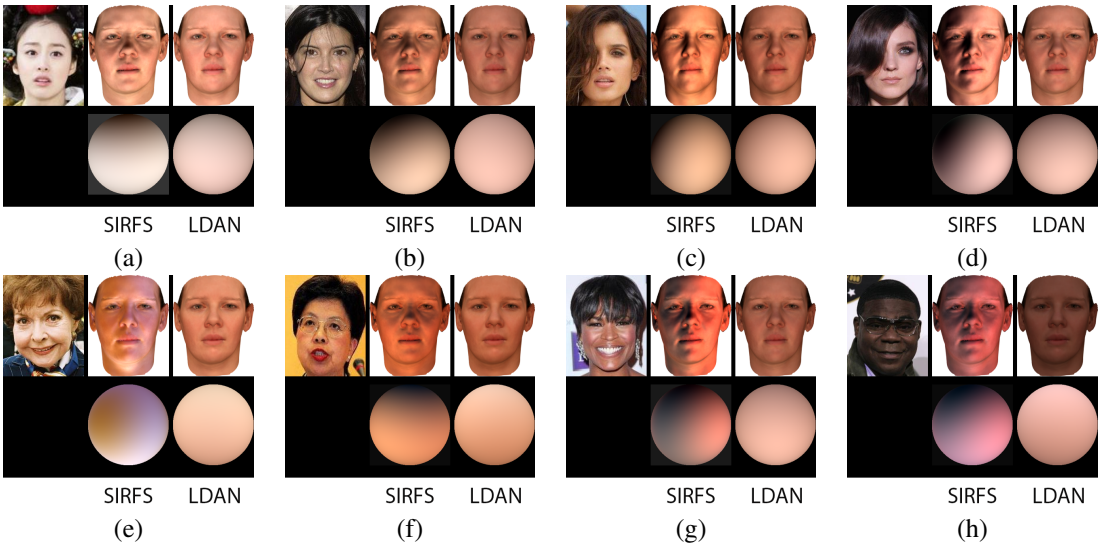

Figure 5. First row: CelebA face image, rendered synthetic face with SIRFS estimated lighting, and rendered synthetic face with LDAN estimated lighting. Second row: the hemisphere visualization of the corresponding estimated lightings. Images are best viewed on screen.

networks. Different from 3DINN, we formulate keypoint detection as a regression problem. Similar to our LDAN model, the network is designed to have a feature network and regression network [1]. We first use the synthetic data provided by [25] to train a keypoint regression network using Equation (1). Since we do not have two sofa images that have exactly the same 2D keypoints, we ignore the feature loss. Then we train the feature network for real sofa data provided by 3DINN using Equation (2).

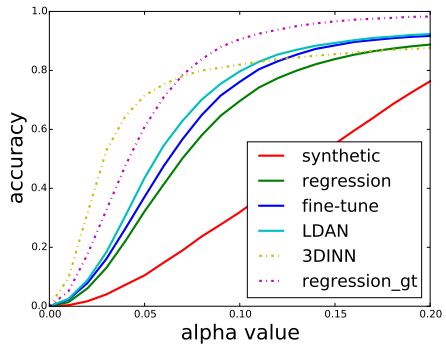

Figure 6. PCK curve of real sofa images for different methods.

We use the Percentage of Correct Keypoints (PCK) metric [44] to evaluate the accuracy. A 2D keypoint prediction is correct if it lies within a radius $\alpha * L$ of the ground truth, where L is the diagonal of the image with $0 < \alpha < 1$. Following [42], we show the PCK curves of LDAN and several baselines with the value of α between 0.0 and 0.2 in Figure 6: (1) Training the network using real data with the noisy label ("regression"); (2) Training the network using synthetic data and testing it on real data ("synthetic"); (3) Fine tune the network trained on synthetic data using real images with the noisy labels ("fine-tune"). We also show the performance of 3DINN and performance of training the network using ground truth label without noise as references ("3DINN" and "regression_gt", both are trained with real ground truth). At $\alpha = 0.1$, LDAN's PCK value (79.66%) outperforms both "regression" (69.61%) and "fine-tune" (76.12%). This shows that LDAN works better than other methods that trained with noisy labels.

[1] See the supplementary material for the details of the network structure.

5. Conclusion

In this paper, we propose a lighting regression network to predict Spherical Harmonics of environment lighting from face images. Lacking the ground truth labels for real face images, we applied an existing method to get noisy ground truth. To alleviate the effect of noise, we propose to apply the idea of adversarial networks and use synthetic face images with known ground truth to help train a deep CNN for lighting regression. Compared with existing methods, the proposed method is more efficient and could predict more consistent Spherical Harmonics from different faces taken under the same environment. We further apply the proposed method to regress 2D keypoint, for which ground truth labels are provided. Our experiments further demonstrate the effectiveness of the proposed method.

6. Acknowledgements

This work was supported by NSF grants (IIS-1302338 and 1526234) and the DARPA MediFor program under cooperative agreement FA87501620191, "Physical and Semantic Integrity Measures for Media Forensics".

References

[1] M. Abadi, A. Agarwal, P. Barham, et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] O. Aldrian and W. A. P. Smith. Inverse rendering of faces with a 3d morphable model. *IEEE Transactions on PAMI*, 35(5), 2013.

[3] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein GAN. *ArXiv e-prints*, abs/1701.07875, 2017.

[4] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Robust discriminative response map fitting with constrained local models. In *CVPR*, 2013.

[5] S. Azadi, J. Feng, S. Jegelka, and T. Darrell. Auxiliary image regularization for deep cnns with noisy labels. In *ICLR*, 2016.

[6] J. T. Barron and J. Malik. Shape, illumination, and reflectance from shading. *IEEE Transactions on PAMI*, 37(8), 2015.

[7] R. Basri, D. Jacobs, and I. Kemelmacher. Photometric stereo with general, unknown lighting. *IJCV*, 72(3), 2007.

[8] R. Basri and D. W. Jacobs. Lambertian reflectance and linear subspaces. *IEEE Transactions on PAMI*, 25(2), 2003.

[9] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH*, 1999.

[10] J. Booth, E. Antonakos, S. Ploumpis, G. Trigeorgis, Y. Panagakis, and S. Zafeiriou. 3d face morphable models "in-the-wild". In *CVPR*, 2017.

[11] F. Chollet et al. Keras. `https://github.com/fchollet/keras`, 2015.

[12] B. Egger, S. Schönborn, A. Schneider, A. Kortylewski, A. Morel-Forster, C. Blumer, and T. Vetter. Occlusion-aware 3d morphable models and an illumination prior for face image analysis. *IJCV*, 2018.

[13] B. Frnay and A. Kaban. A comprehensive introduction to label noise. In *ESANN*, 2014.

[14] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, 2015.

[15] T. Gerig, A. Forster, C. Blumer, B. Egger, M. Lüthi, S. Schönborn, and T. Vetter. Morphable face models - an open framework. *ArXiv e-prints*, abs/1709.08398, 2017.

[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*.

[17] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image Vision Computing*, 28(5), 2010.

[18] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[19] M. Heredia Conde, D. Shahlaei, V. Blanz, and O. Loffeld. Efficient and robust inverse lighting of a single face image using compressive sensing. In *ICCV Workshops*, 2015.

[20] G. Hinton, N. Srivastava, and K. Swersky. Lecture 6a, overview of mini-batch gradient descent. `http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf`, 2012.

[21] I. Jindal, M. Nokleby, and X. Chen. Learning deep networks from noisy labels with dropout regularization. In *ICDM*, 2016.

[22] M. K. Johnson and H. Farid. Exposing digital forgeries in complex lighting environments. *IEEE Transactions on IFS*, 2(3), 2007.

[23] H. Kim, M. Zollhöfer, A. Tewari, J. Thies, C. Richardt, and C. Theobalt. Inversefacenet: Deep single-shot inverse face rendering from A single image. *ArXiv e-prints*, abs/1703.10956, 2017.

[24] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. Tenenbaum. Deep convolutional inverse graphics network. In *NIPS*.

[25] C. Li, Z. Zia, Q. huy Tran, X. Yu, G. D. Hager, and M. Chandraker. Deep supervision with shape concepts for occlusion-aware 3d object parsing. In *CVPR*, 2017.

[26] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

[27] V. Mnih and G. E. Hinton. Learning to label aerial images from noisy data. In *ICML*, 2012.

[28] G. Patrini, A. Rozza, A. Menon, R. Nock, and L. Qu. Making deep neural networks robust to label noise: a loss correction approach. In *CVPR*, 2017.

[29] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3d face model for pose and illumination invariant face recognition. In *AVSS*, 2009.

[30] B. Peng, W. Wang, J. Dong, and T. Tan. Optimized 3d lighting environment estimation for image forgery detection. *IEEE Transactions on IFS*, 12(2), 2017.

[31] R. Ramamoorthi. Analytic pca construction for theoretical analysis of lighting variability in images of a lambertian object. *IEEE Transactions on PAMI*, 24(10), 2002.

[32] R. Ramamoorthi and P. Hanrahan. On the relationship between radiance and irradiance: Determining the illumination from images of a convex lambertian object. *JOSA*, 2001.

[33] K. Saito, Y. Mukuta, Y. Ushiku, and T. Harada. Deep modality invariant adversarial network for shared representation learning. In *ICCV*, 2017.

[34] S. Sankaranarayanan, Y. Balaji, C. D. Castillo, and R. Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. *ArXiv e-prints*, abs/1704.01705, 2017.

[35] D. Shahlaei and V. Blanz. Realistic inverse lighting from a single 2d image of a face, taken under unknown and complex lighting. In *FG*, 2015.

[36] Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, and D. Samaras. Neural face editing with intrinsic image disentangling. In *CVPR*, 2017.

[37] S. Sukhbaatar, J. Bruna, M. Paluri, L. Bourdev, and R. Fergus. Learning from noisy labels with deep neural networks. In *ICLR*, 2015.

[38] Y. Tang, R. Salakhutdinov, and G. Hinton. Deep lambertian networks. In *ICML*, 2012.

[39] A. Tuan Tran, T. Hassner, I. Masi, and G. Medioni. Regressing robust and discriminative 3d morphable models with a very deep neural network. In *CVPR*, 2017.

[40] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *CVPR*, 2017.

[41] Y. Wang, L. Zhang, Z. Liu, G. Hua, Z. Wen, Z. Zhang, and D. Samaras. Face relighting from a single image under arbitrary unknown lighting conditions. *IEEE Transactions on PAMI*, 31(11), 2009.

[42] J. Wu, T. Xue, J. J. Lim, Y. Tian, J. B. Tenenbaum, A. Torralba, and W. T. Freeman. Single image 3d interpreter network. In *ECCV*, 2016.

[43] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang. Learning from massive noisy labeled data for image classification. In *CVPR*, 2015.

[44] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011.

[45] M. D. Zeiler. ADADELTA: an adaptive learning rate method. *ArXiv e-prints*, abs/1212.5701, 2012.

[46] L. Zhang and D. Samaras. Face recognition from a single training image under arbitrary unknown lighting using spherical harmonics. *IEEE Transaction on PAMI*, 28(3), 2006.