Deep Feature Interpolation

(T-PAMI 2021, CVPR 2017)

Deep Feature Interpolation for Image Content Changes

Paul Upchurch[1,*] Jacob Gardner[1,*] Geoff Pleiss[1] Robert Pless[2] Noah Snavely[1] Kavita Bala[1]
Kilian Weinberger[1]

[1]Cornell University
[2]George Washington University
[*]Authors contributed equally

(CVPR, 2017)

Deep Feature Interpolation (DFI)

Input Older

Figure 1. Aging a face with DFI.

Deep Feature Interpolation (DFI)

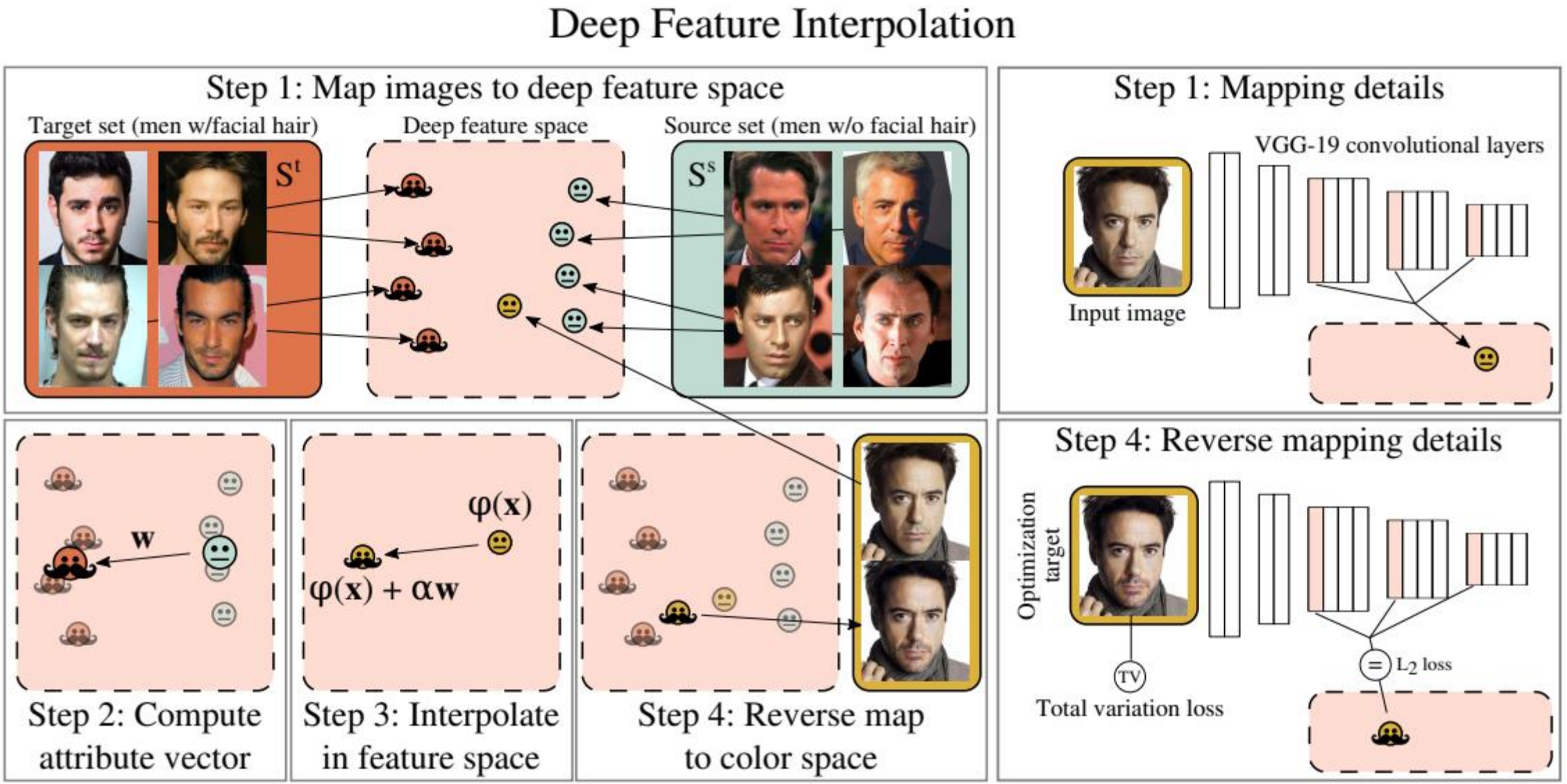


Figure 2. A schematic outline of the four high-level DFI steps.

Four high-level steps

- Map the images in the target S^t and source S^s set into the deep feature representation through the pre-trained (VGG-19) conv-net.

- Compute the mean feature values for each set of images, $\bar{\phi}^t$ and $\bar{\phi}^s$, and define their differences as the attribute vector.

$$w = \bar{\phi}^t - \bar{\phi}^s$$

- Map the test image x to a point $\phi(x)$ in the deep feature space and move it along the attribute vector w, resulting in $\phi(x) + a\mathrm{w}$.

- Reconstruct the transformed output image z by solving the reverse mapping into pixel space.

$$\phi(z) = \phi(x) + aw$$

Detail

- Selecting S^t and S^s.

These neighbors can be selected in two ways.
The attribute labels are (un)available.

$$\bar{\phi}^t = \frac{1}{K} \sum_{\mathbf{x}^t \in \mathcal{N}_K^t} \phi(\mathbf{x}^t) \text{ and } \bar{\phi}^s = \frac{1}{K} \sum_{\mathbf{x}^s \in \mathcal{N}_K^s} \phi(\mathbf{x}^s). \quad (3)$$

- Deep feature mapping.
  - VGG19 pre-trained on ILSVRC2012, which has proven to be effective at artistic style transfer.
  - Pick the first layer from the last three regions, conv3_1, conv4_1 and conv5_1.

- Reverse mapping.

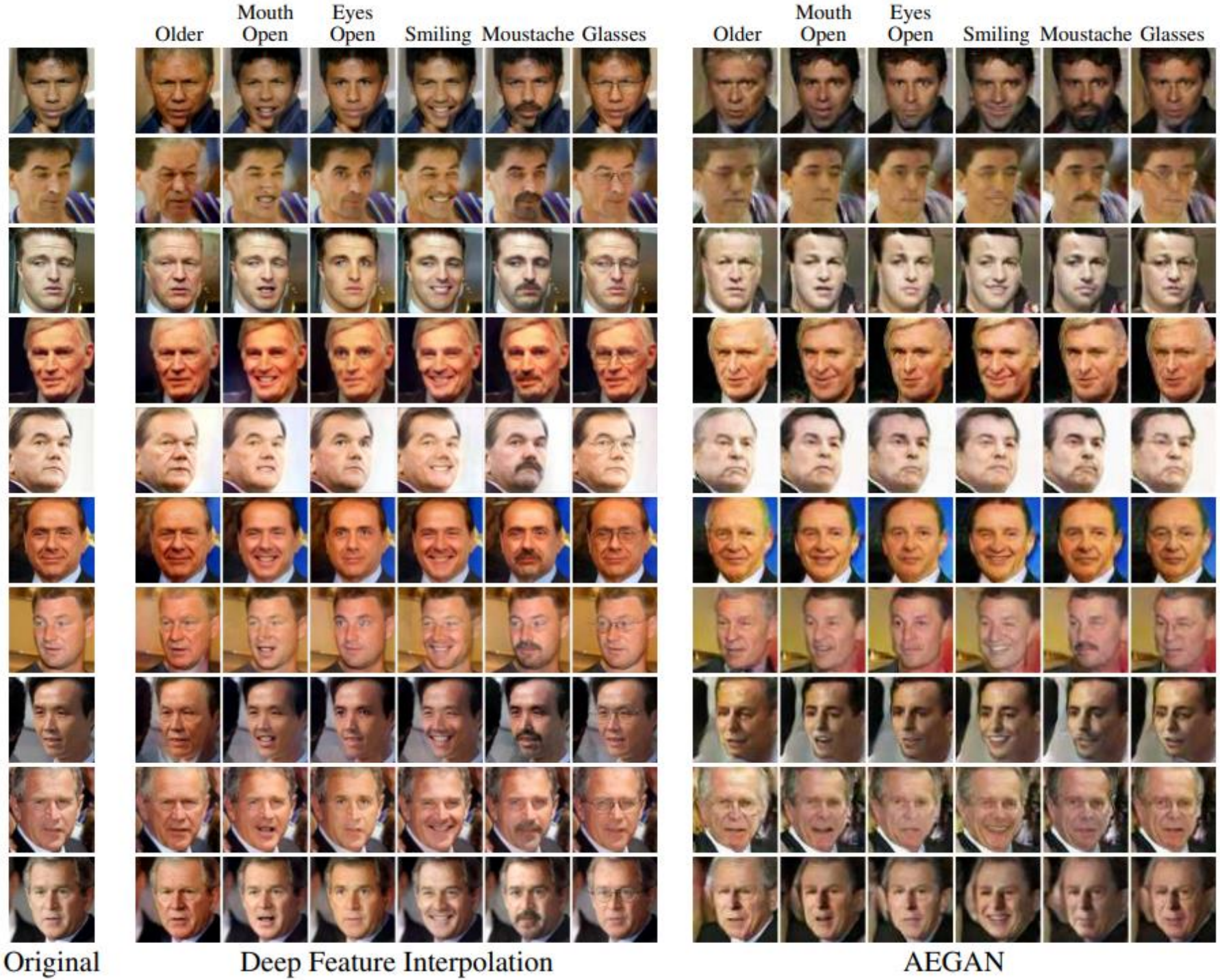
$$\mathbf{z} = \arg\min_{\mathbf{z}} \frac{1}{2} \|(\phi(\mathbf{x}) + \alpha \mathbf{w}) - \phi(\mathbf{z})\|_2^2 + \lambda_{V^\beta} R_{V^\beta}(\mathbf{z}), \quad (4)$$

where R_{V^β} is the Total Variation regularizer [28] which encourages smooth transitions between neighboring pixels,

$$R_{V^\beta}(\mathbf{z}) = \sum_{i,j} \left((z_{i,j+1} - z_{i,j})^2 + (z_{i+1,j} - z_{i,j})^2 \right)^{\frac{\beta}{2}} \quad (5)$$

Here, $z_{i,j}$ denotes the pixel in location (i, j) in image $\mathbf{z}$.

Experiments



Experiments

Figure 4. **(Zoom in for details.)** Filling missing regions. **Top.** LFW faces. **Bottom.** UT Zappos50k shoes. Inpainting is an interpolation from masked to unmasked images. Given any dataset we can create a source and target pair by simply masking out the missing region. **DFI uses** $K = 100$ such pairs derived from the nearest neighbors (excluding test images) in feature space. The face results match wrinkles, skin tone, gender and orientation (compare noses in 3rd and 4th images) but fail to fill in eyeglasses (3rd and 11th images). The shoe results match style and color but exhibit silhouette ghosting due to misalignment of shapes. Supervised attributes were not used to produce these results. For the curious, we include the source image but we note that the goal is to produce a plausible region filling—not to reproduce the source.

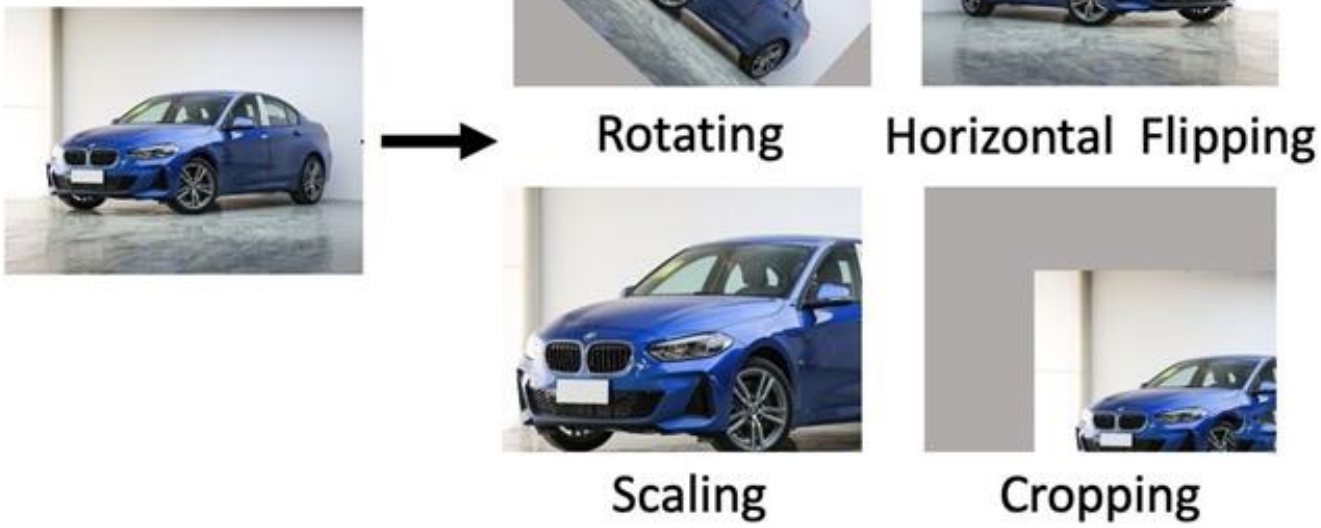
Regularizing Deep Networks with Semantic Data Augmentation

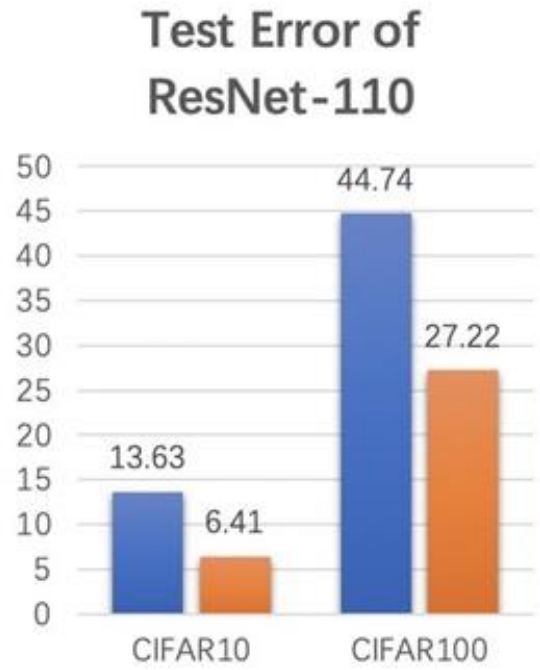
Yulin Wang*, Gao Huang*, *Member, IEEE*, Shiji Song, *Senior Member, IEEE*, Xuran Pan, Yitong Xia, and Cheng Wu

(T-PAMI, 2021)

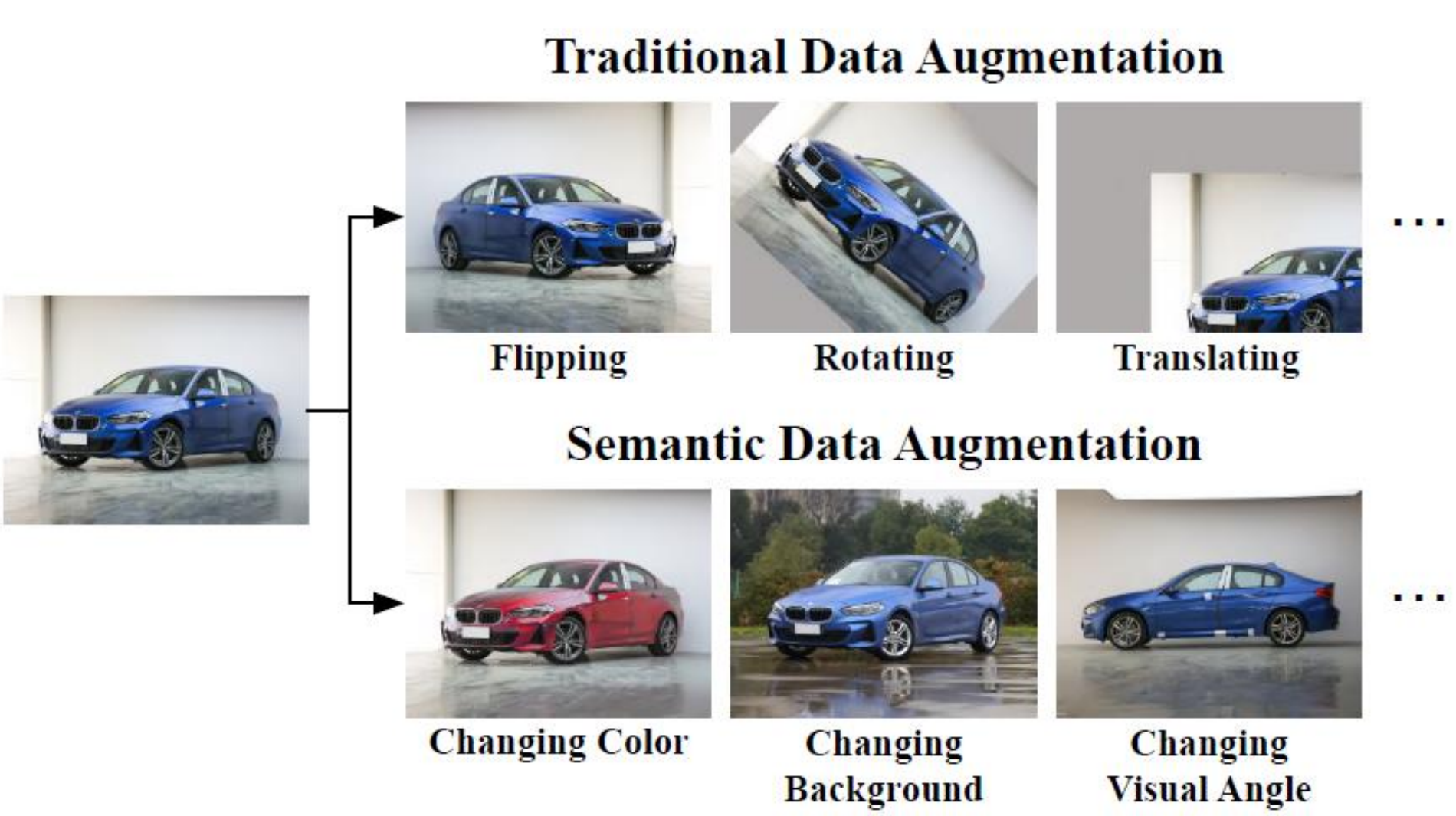
Traditional data augmentation

- Traditional data augmentation is an effective technique to alleviate the overfitting problem in training deep networks.





Traditional VS. Semantic data augmentation



Data augmentation by GAN

数据少->GAN难训练->扩增效果差

数据多->GAN训练好->扩增难以提供效果

😄 ✓ Semantic Augmentation.

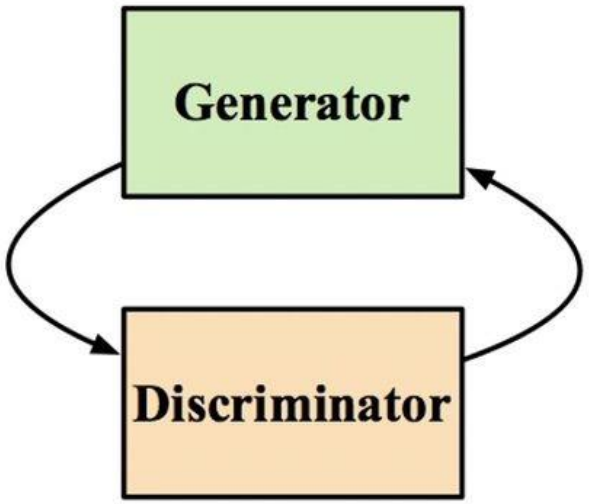
✓ Intuitive.

😔 ✗ Complex.

✗ Inefficient.

✗ Marginal Improvement.

Training Data



Pre-trained GAN

……

Idea

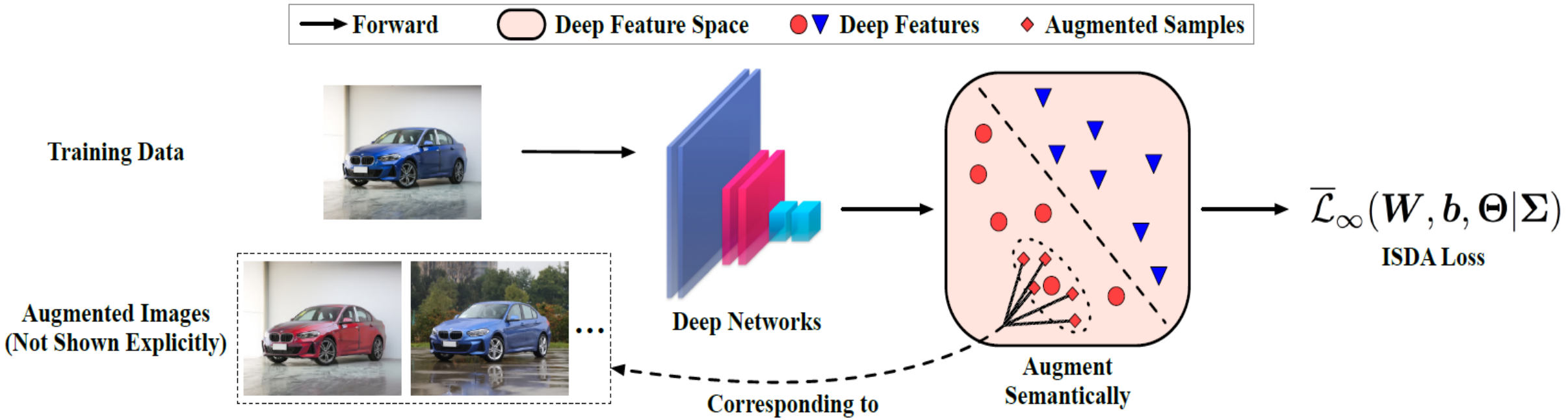
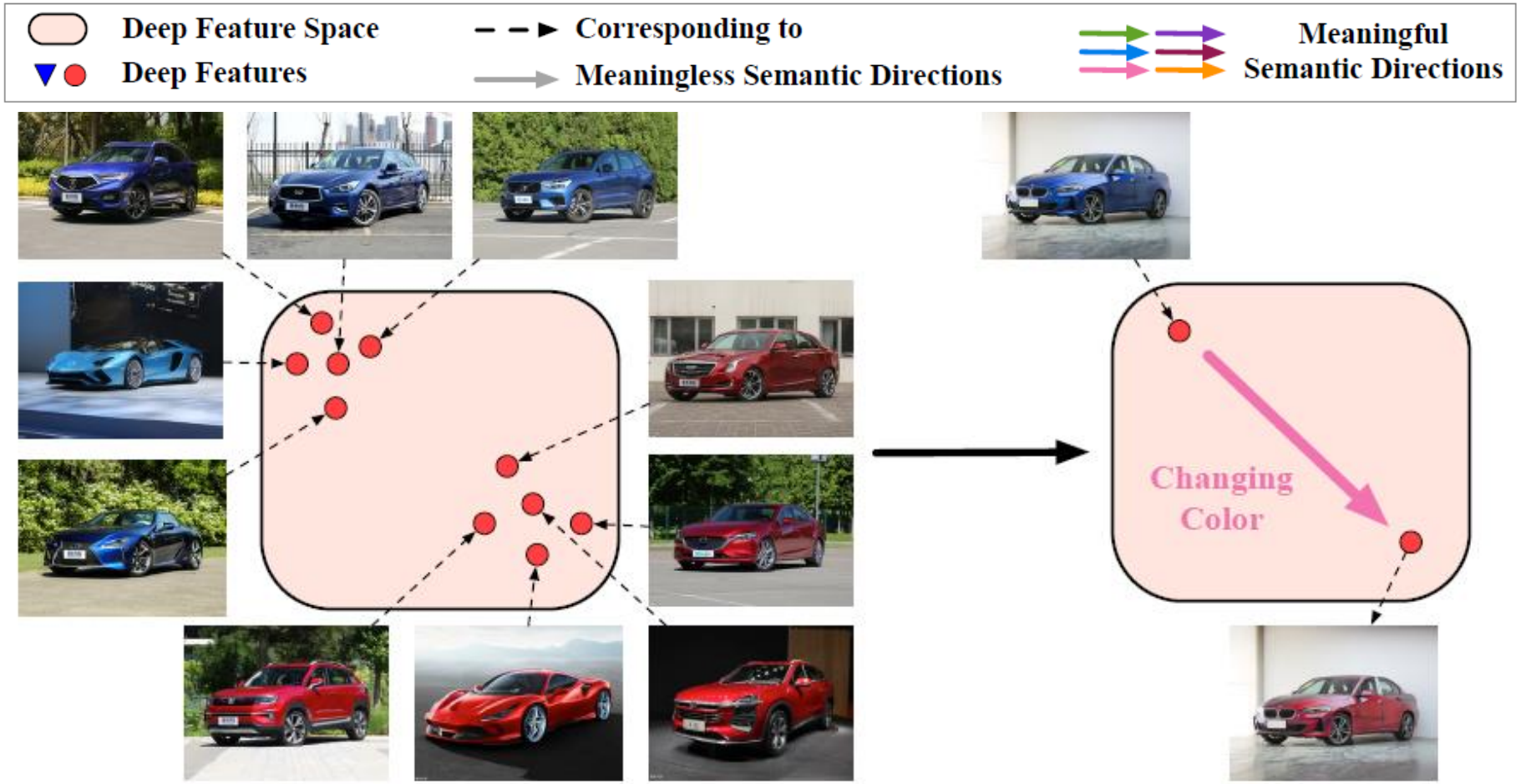


Fig. 2. An overview of ISDA. Inspired by the observation that certain directions in the feature space correspond to meaningful semantic transformations, we augment the training data semantically by translating their features along these semantic directions, without involving auxiliary deep networks. The directions are obtained by sampling random vectors from a zero-mean normal distribution with dynamically estimated class-conditional covariance matrices. In addition, instead of performing augmentation explicitly, ISDA boils down to minimizing a closed-form upper-bound of the expected cross-entropy loss on the augmented training set, which makes our method highly efficient.

Motivation

- 卷积网络一个有趣的性质：由于我们用线性分类器约束网络输出，深度网络的特征往往是**线性化**的，输入空间中不同样本之间复杂的语义关系倾向于表现为其对应深度特征之间的**简单线性关系**。换言之，**深度特征空间中的一些方向是对应特定语义变换的**。

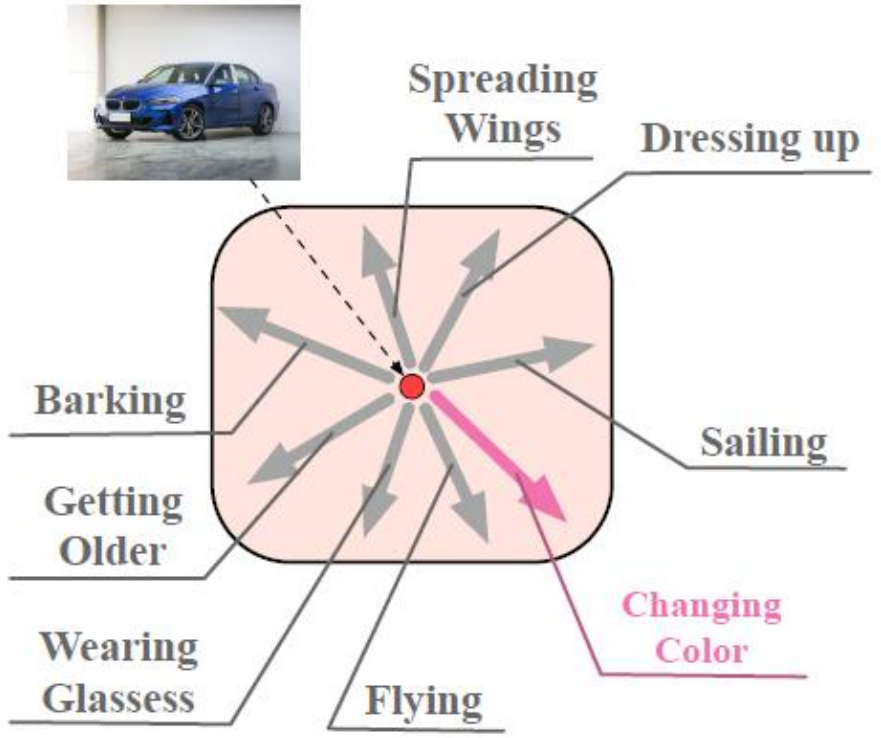


(a) Human Annotation

Method

- Human annotation X
  - Huge annotation cost.
  - It is difficult to pre-define all possible semantic transformations for each class.

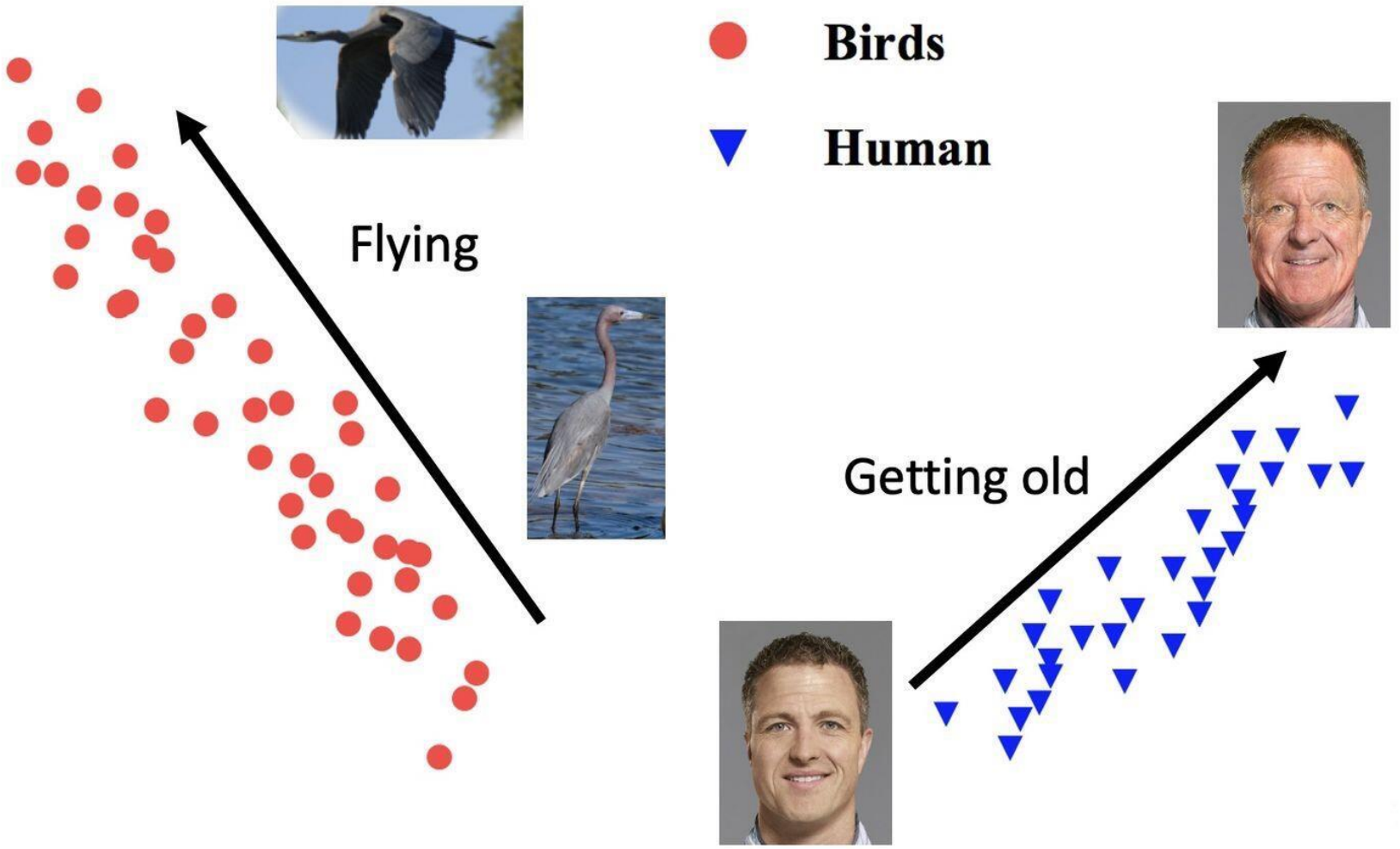
- Random sampling X
  - Sampling totally at random will yield many meaningless semantic directions.



(b) Random Sampling

Methods – Semantic Directions Sampling

- Each category of samples has its own distribution. In fact, this data distribution implies the potential semantic direction. "Bird" has a large variance in the direction of "flying", while variance in the direction of "getting old" is almost 0.

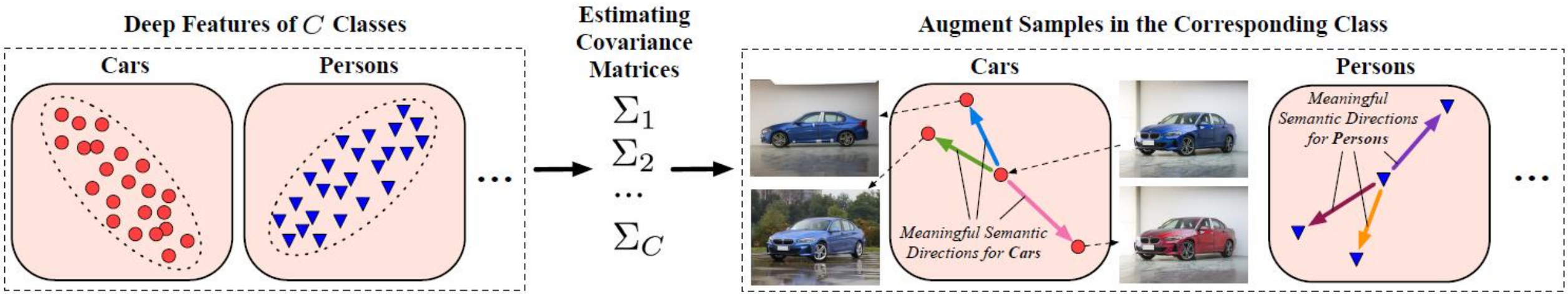


Like PCA

对某一类样本的深度特征在其最大方差方向上进行PCA降维，得到的是其语义信息的补全。

Methods – Semantic Directions Sampling

- 通过统计每一类别的类内协方差矩阵，为每一类别构建了一个零均值的高斯分布，进而从中采样出有意义的语义变换方向，用于各自类别内的数据扩增。



(c) Class-Conditional Gaussian Sampling

$$\tilde{\boldsymbol{a}}_i \sim \mathcal{N}(\boldsymbol{a}_i, \lambda\Sigma_{y_i}),$$

- Augmented features
- Coefficient
- Covariance matrix
- Original deep features

Methods

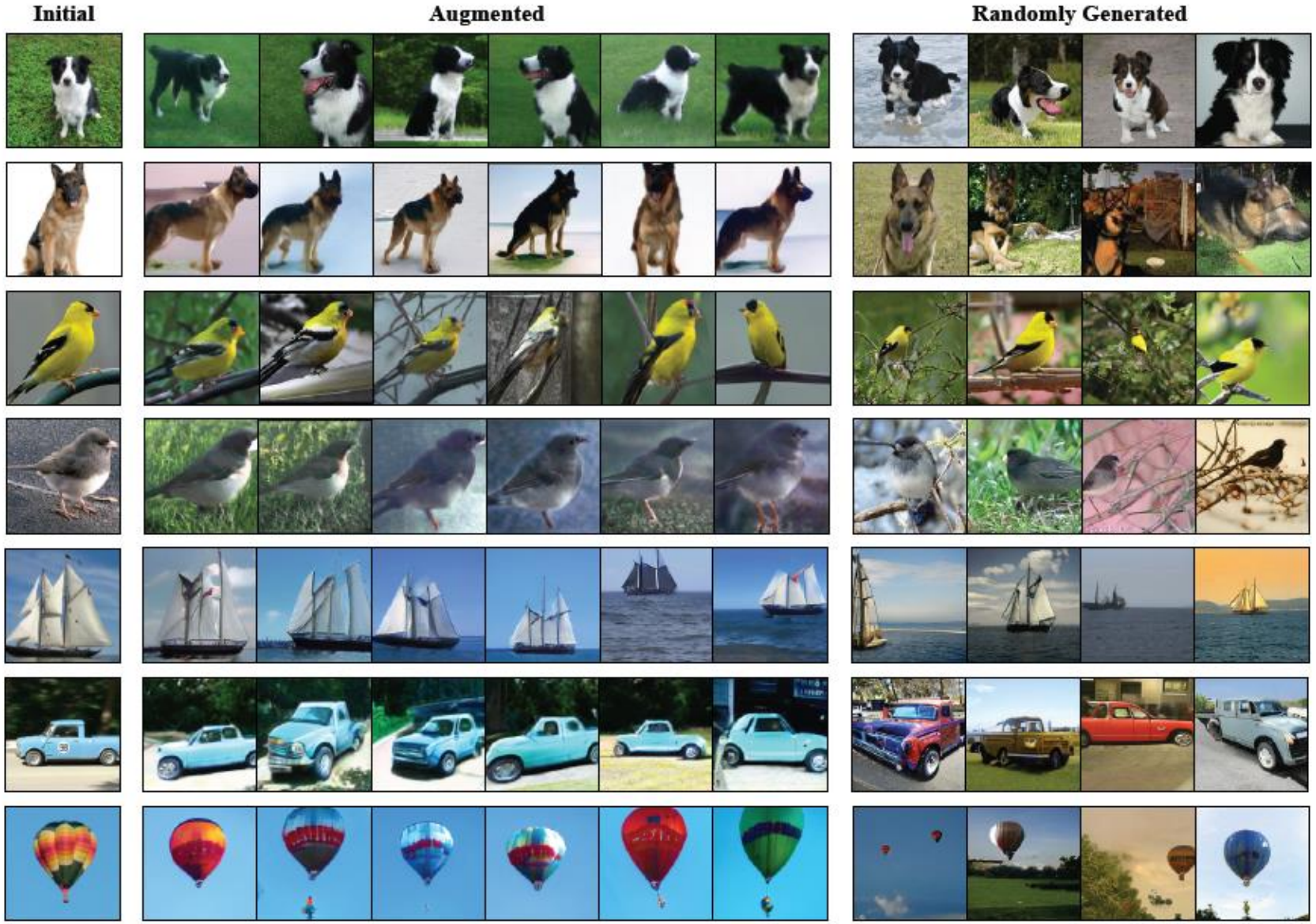
Algorithm 1 The ISDA algorithm.

Input: $\mathcal{D}$, λ_0
Randomly initialize $\boldsymbol{W}$, $\boldsymbol{b}$ and $\boldsymbol{\Theta}$
for $t = 0$ **to** T **do**
  Sample a mini-batch $\{\boldsymbol{x}_i, y_i\}_{i=1}^{B}$ from $\mathcal{D}$
  Compute $\boldsymbol{a}_i = G(\boldsymbol{x}_i, \boldsymbol{\Theta})$
  Estimate the covariance matrices Σ_1, Σ_2, ..., Σ_C
  Compute $\overline{\mathcal{L}}_\infty$ according to Eq. (7)
  Update $\boldsymbol{W}$, $\boldsymbol{b}$, $\boldsymbol{\Theta}$ with SGD
end for
Output: $\boldsymbol{W}$, $\boldsymbol{b}$ and $\boldsymbol{\Theta}$

$$\mathcal{L}_\infty(\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\Theta}|\boldsymbol{\Sigma}) = \frac{1}{N}\sum_{i=1}^{N} \mathrm{E}_{\tilde{\boldsymbol{a}}_i}[-\log(\frac{e^{\boldsymbol{w}_{y_i}^{\mathrm{T}}\tilde{\boldsymbol{a}}_i + b_{y_i}}}{\sum_{j=1}^{C} e^{\boldsymbol{w}_j^{\mathrm{T}}\tilde{\boldsymbol{a}}_i + b_j}})]. \quad (6)$$

$$\mathcal{L}_M(\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{\Theta}) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{M}\sum_{m=1}^{M} -\log(\frac{e^{\boldsymbol{w}_{y_i}^{\mathrm{T}}\boldsymbol{a}_i^m + b_{y_i}}}{\sum_{j=1}^{C} e^{\boldsymbol{w}_j^{\mathrm{T}}\boldsymbol{a}_i^m + b_j}}), \quad (5)$$

Experiments



BigGAN

Fig. 7. Visualization of the semantically augmented images on ImageNet. ISDA is able to alter the semantics of images that are unrelated to the class identity, like backgrounds, actions of animals, visual angles, etc. We also present the randomly generated images of the same class.

Thanks